CSTDM09 - California Statewide Travel Demand Model

Model Development

User Guide

Final System Documentation: Technical Note

ULTRANS
Institute of Transportation Studies, UC Davis
Davis, California

HBA Specto Incorporated
Calgary, Alberta

May 2011

Table of Contents

Tables:

Figures:

1. Introduction

This User Guide is part of the documentation for the 2009 California Statewide Travel Demand Model (CSTDM09). The User Guide describes how the model is applied in the CUBE software environment to produce travel demand forecasts and resulting network loading outputs for a particular scenario. It is intended to serve as a reference document for travel modeling personnel using the model.

The complete CSTDM09 documentation includes the 19 documents reported in Table 1.

Table 1: Overview of CSTDM09 Documentation

Document	**File Name**
1. Model Overview	*CSTDM09_ModelOverview_Final.pdf*
2. Transportation Analysis Zones	*CSTDM09_TAZ_LUZ_Final.pdf*
3. Network Preparation and Coding	*CSTDM09_Networks_Final.pdf*
4. Local Transit Functions	*CSTDM09_Local_Transit_Function_Final.pdf*
5. Zonal Properties	*CSTDM09_ZonalProperties_Final.pdf*
6. Population	*CSTDM09_Population_Final.pdf*
7. Employment	*CSTDM09_Employment_Final.pdf*
8. Short Distance Personal Travel Model (SDPTM) Part 1	*CSTDM09_SDPTM_Part1_Final.pdf*
9. Short Distance Personal Travel Model (SDPTM) Part 2	*CSTDM09_ SDPTM_Part2_Final.pdf*
10. Short Distance Personal Travel Model (SDPTM) Part 3	*CSTDM09_ SDPTM_Part3_Final.pdf*
11. Short Distance Commercial Vehicle Model (SDCVM)	*CSTDM09_SDCVM_Final.pdf*
12. Long Distance Personal Travel Model (LDPTM)	*CSTDM09_LDPTM_Final.pdf*
13. Long Distance Commercial Vehicle Model (LDCVM)	*CSTDM09_LDCVM_Final.pdf*
14. External Travel Model (ETM)	*CSTDM09_ETM_Final.pdf*
15. School Enrollment	*CSTDM09_SchoolEnrollment_Final.pdf*
16. Parking Costs	*CSTDM09_ParkingCost_Final.pdf*
17. Travel Behavior Datasets	*CSTDM09_TravelBehavior_Final.pdf*
18. Model Validation	*CSTDM09_Validation_Final.pdf*
19. User guide	*CSTDM09_UserGuide_Final.pdf*

For additional information on the CSTDM travel model components, please refer to the documents reported in Table 1, which describe the individual model components in more detail.

The User Guide assumes a basic knowledge of the CUBE package of programs for travel model operation (CUBE Base, CUBE Voyager and CUBE Cluster), plus the ability to manipulate data in database formats.

2. Model Overview

The 2009 California Statewide Travel Demand Model forecasts all personal travel made by every California resident, plus all commercial vehicle travel, made on a typical weekday in the Fall / Spring (when schools are in session). It has five demand models:

- A Short Distance Personal Travel Model (for intra-California trips) (SDPTM);
- A Long Distance Personal Travel Model (for intra-California trips) (LDPTM);
- A Short Distance Commercial Vehicle Model (for intra-California trips) (SDCVM);
- A Long Distance Commercial Vehicle Model (for intra-California trips) (LDCVM);
- An External Vehicle Trip Model (for trips with an origin and/or destination outside California).

The state is subdivided into 5,191 traffic analysis zones (TAZs). In addition, there are 48 external zone vehicle entry / exit points on roads on the state boundary; plus three external zone seaports whose import / export activities generate significant truck activity (Long Beach, Los Angeles and Oakland). The zones nest both within the 58 California counties and the 524 land use zone (LUZ) system used in the California PECAS spatial economic model. Figure 1 illustrates the CSTDM TAZ system:

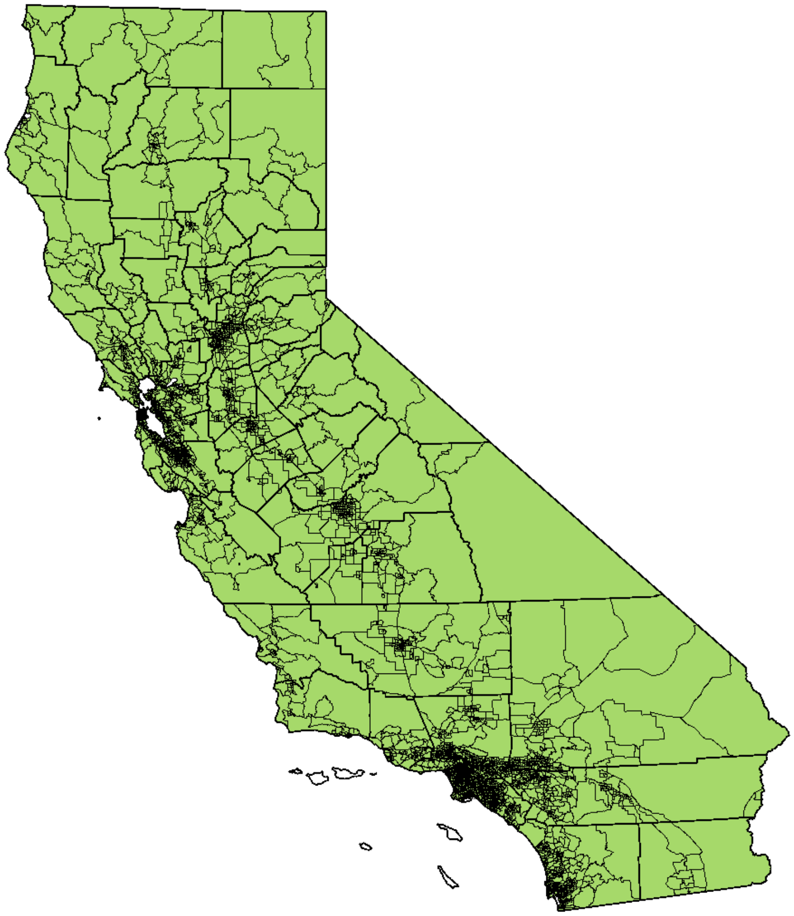


Figure 1: TAZ system

The cut-off distance between short and long distance personal travel models is 100 miles (defined by the straight-line distance between TAZ centroids). This 100-mile definition is consistent with that used for the California High Speed Rail Travel Model for person trips. All TAZ-to-TAZ personal travel movements within 100 miles are forecast by the SDPTM; and all TAZ-to-TAZ personal travel movements 100 miles and longer are forecast by the LDPTM.

The cut-off distance between short and long distance commercial vehicle models is 50 miles (defined by the straight-line distance between TAZ centroids). This 50-mile definition is consistent with the depot spacing for commercial shippers.

All TAZ-to-TAZ commercial vehicle movements within 50 miles are forecast by the SDCVM; and all TAZ-to-TAZ commercial movements 50 miles and longer are forecast by the LDCVM.

The External Vehicle Trip model forecasts car and commercial vehicle trips made between the 51 external zones and the 5,191 internal TAZs.

The five demand models use as inputs:

- Demographic data for each TAZ (population and household characteristics; employment by industry and occupation; school enrolment);
- Other zonal attributes (area; area type; population and population density; parking costs; region);
- Travel cost data (fuel costs; public transit fares; road tolls);
- Commodity flow movements (for the LDCVM);
- Observed Vehicle flows and growth factors by type and time period (for the External Vehicle Trip model);
- TAZ to TAZ travel times and costs by mode and time period (obtained from the CUBE network descriptions referenced below).

The weekday time-frame of the models is split into four time periods for demand modeling and travel assignment purposes:

- An AM Peak Period (6AM to 10AM);
- A Midday Period (10AM to 3PM);
- A PM Peak Period (3PM to 7PM);
- An Offpeak Period (12AM to 6AM plus 7PM to midnight).

The demand models generally further sub-divide the Offpeak period into an Early time period and a Late time period. The Early period is defined as being between 3AM and 6AM; and the Late time period as being between 7PM and 3AM. These definitions are consistent with the data collection approach for household travel surveys, where the travel survey day is defined as starting at 3AM. For the purposes of the model assignment and skim extraction, this subdivision is not considered, and both Early time period and Late time are always considered in the aggregated Offpeak Period.

Each of the travel demand sub-models considers a different set of travel modes relevant to their travel type.

- The SDPTM considers 8 travel modes – Single Occupant Car (SOV); High Occupant Car with 2-persons in the car (HOV2); High Occupant Car with 3+persons in the car (HOV3); Walk Access Local Transit (bus, light rail, heavy rail); Drive Access Local Transit (where access to or egress from a rail station is by car); Walk; Bicycle; School Bus.
- The LDPTM considers 4 travel modes – Car; Air; Conventional Rail; High speed Rail.
- The SDCVM considers 3 commercial vehicle types – Light commercial vehicle; Medium Truck; Heavy Truck.
- The LDCVM considers 1 commercial vehicle type – Heavy Truck.
- The External vehicle Trip model considers 5 travel modes – Single Occupant Car (SOV); High Occupant Car with 2-persons in the car (HOV2); High Occupant Car with 3+persons in the car (HOV3); Medium Truck; Heavy Truck.

Road network descriptions for each time period are coded in the standard CUBE format. For the road network, all freeway, expressway and most arterial roadways are explicitly represented, with collector and local roads mostly covered through zone centroid connector links. Link distances, free flow speeds and capacities are explicitly coded. Figure 2 illustrates the year 2000 road network coding – the overall network contains over 86,000 nodes and 235,000 links[1].

[1] Policy scenarios involving modifications in the road and public transit networks can be created in a rather straightforward way in CSTDM. Road networks are modified in CSTDM through conventional CUBE commands. Appendix A provides some guidance on simple CUBE commands used for network maintenance and modifications.

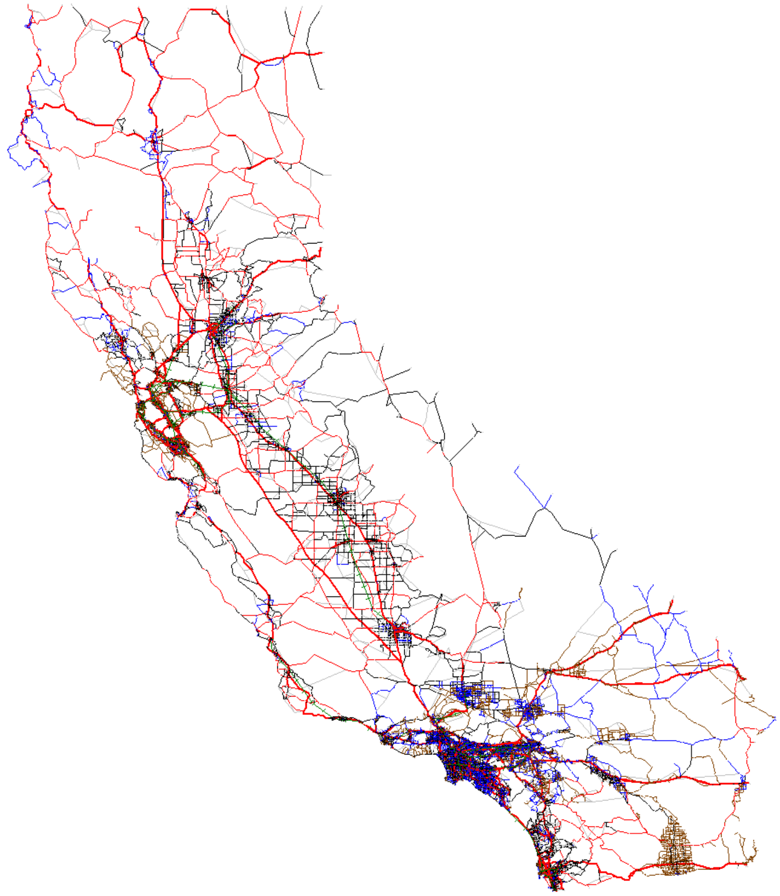


Figure 2: Base Road Network

For public transit, all air and rail lines and services are explicitly coded using the standard CUBE format. For local bus transit, a simplified model is used to give level of service times and costs, based on road network speeds, land use variables, and transit operator service measures. Walk and bicycle times are derived from road network distances.

The forecast vehicle trip tables from each sub-model are grouped, for each of the four main time periods (AM, Midday, PM and Offpeak), and assigned to the road network.

Overall Model output includes:

- TAZ to TAZ travel demands by mode by time period, which can be summed by any geography, for example to give intra-regional and inter-regional flows;

- "Loaded" road networks by time period giving link flows by vehicle type and link operating performance;
- TAZ to TAZ travel times and costs by mode and time period.

This output can be presented graphically. It can also be broken down by person type as required, for equity considerations.

Figure 3 summarizes the CSTDM model system operation.

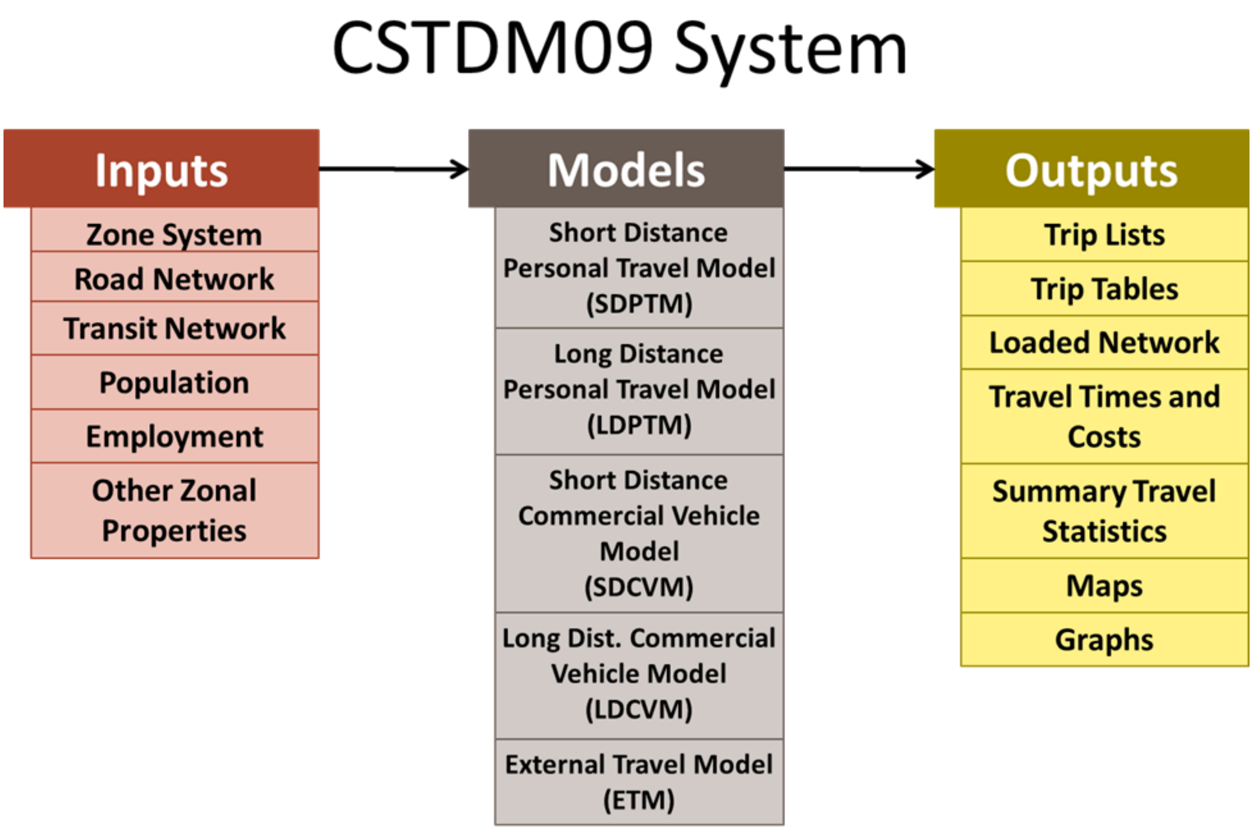


Figure 3: Overall CSTDM Operation

The overall model is run within the CUBE program environment, with the following major steps:

1. Prepare all required input data.

2. Run CUBE scripts to assign vehicle trip tables, and obtain initial TAZ to TAZ travel skims (times and costs) for each mode for each time period. (In practice skims from a previous model run may be available for use, and this step can be omitted).
3. Run each of the Demand models separately, using the input data from Step 1 and the initial travel skim inputs from Step 2.
4. Combine the resulting vehicle trip table outputs by each vehicle type and time period, and run CUBE scripts to assign vehicle trip tables, and obtain updated TAZ to TAZ travel skims (times and costs) for each mode for each time period.
5. Run each of the Demand models separately, using the input data from Step 1 and the revised travel skim inputs from Step 4.
6. Repeat Steps 4 and 5 in an iterative process until model convergence is reached. Convergence is achieved when the forecasts of travel demand and associated travel times and costs do not differ significantly between successive iterations of the entire model. The current implementation of the CSTDM runs the demand model / travel assignment and skims process 3 times, for this convergence process. This limit on the number of iterations is required because of the need to put a limit on computer run times for the demand models and trip assignment (see below).
7. Produce required outputs from the final run of the model.

3. Hardware Requirements

CSTDM is a very complex and computationally intensive application. It is essential that it be run on a very powerful computer. The user experience on any desktop available as of the release of the model is unlikely to be satisfactory. A server is the recommended physical platform.

The CSTDM was developed on a server with the following specifications:

- Manufacturer: Silicon Mechanics
- Processors: 2 Intel E5520 (Quad core) processors with a 2.27GHz clock speed

- RAM: 24 Giga-Bytes
- Hard Drives: 8 Tera-Bytes
- Operating System: Windows Server 2008 Enterprise, Service Pack 2, with Client Access Licenses for each user.

These are not technically minimum requirements for CSTDM, but should be viewed as such because the run times that would result from a lower system specification will not be practical for use.

The operating system must be a 64Bit Windows operating system. The CSTDM should work on 64Bit versions of Windows XP, Windows 7, Windows Server 2008 or a more recent Microsoft Windows Operating system supported by Citilabs, Java and Python.

The CSTDM is distributed with settings appropriate to a computer that has 16 threads available through the CPUs. On the development server this was 2 four core processors each of which used Intel's hyper threading technology to process two threads for a total of 16 threads. Adjustments can be made to adjust the number of threads used. See Section 5.4 for more information on the model run in the multithread environment. Appendix D at the end of this document provides additional information on the adjustments required to run the model in case of hardware specifications with a different number of threads.

4. Software Requirement, Installation, and Basic Operation

The CSTDM makes use of three software environments. CUBE produced by Citilabs Inc. (www.citilabs.com) with its components of Base, Voyager and Cluster handles the vast majority of the highway assignment, public transport, skimming operations, and controls the flows of data between the demand model components running in the free programming languages of Python (www.python.org) and Java (www.java.com). CUBE Cluster allows for multiple CPUs in the server to be used simultaneously to speed up the CUBE based portions of the travel model or to enable the activation of the non-CUBE demand models in parallel with each other.

4.1 Software Requirements

The following software is required to run the CSTDM:

- CUBE (version 5.12): www.citilabs.com
  - Base
  - Voyager
  - Cluster
- Python (version 2.6, 64bit): http://www.python.org/download/releases/2.6.6/
  - Numpy (version 1.4.1): http://numpy.scipy.org/
  - Numexpr (version 1.4): http://code.google.com/p/numexpr/
  - Tables (version 2.2): http://www.pytables.org/
  - dbfpy (version 1.7): http://sourceforge.net/projects/dbfpy/
- Java 6 (Update 18, 64bit): http://www.java.com/en/download/manual.jsp

The following software is recommended for use with the CSTDM.

- Microsoft Excel (for viewing screenline data)
- Adobe Acrobat Reader (for viewing documentation)

In each case where 64 bit software is identified it is important that the 64 bit version of the software be installed, and not a 32 bit version, or the model will fail.

4.2 Installation Instructions

To install the CSTDM, identify space on a disk with a minimum of a Terabyte (1,000 Gigabytes) of free space. This space should be on the same server as the software is installed. Do not try to run this application across a network connection. Doing so will adversely affect run times.

To install the CSTDM, the following steps are required:

1. Install Java (version 6, 64bit) as a standard installation
2. Install Python 2.6 (64bit) as a standard installation
3. Install each of the Python extensions (Numpy, Numexpr, Tables) as a standard installation

4. Install CUBE with Voyager and Cluster, and ensure that the license manager is configured and licensed for the computer that will be running the CSTDM. As of the writing of this manual, all Citilabs products are licensed by the number of CPUs and are tied to a USB based key that enables their operation. Forthcoming versions of Citilabs software may change the licensing system.
5. Copy all of the CSTDM from the distribution source into a directory prepared for its use. In this documentation, we will reference our installation directory, which is: E:\CSTDM2009.

4.3 Running the Model

Running the model can happen in multiple ways depending on the goals. ULTRANS/HBA has generally run the CSTDM in one of four ways:

1. Running the entire model from start to finish with all of its components
2. Running a subsection of the model
3. Running an individual process
4. Running the entire model with the exception of a specific component

In each case, you must select the scenario to be run in the catalog view. As of the writing of this document there are two scenarios: Year 2000 and Year2008.

To run the entire model from start to finish, open the catalog (CSTDM200.cat). Start the Cluster (as described in section 5.2). Then select the

Application menu-->Run Application

The Run Application dialog will appear (Figure 4).

Figure 4: Run Application Dialog

To run the entire model leave all settings as you see here. These are the default settings. When you click OK the application script will be assembled and you will be presented with a second dialog box that either informs you of any errors in the script assembly, or allows you to start the run. Any errors encountered during the compiling are likely to be missing files referenced within the catalog. It is worth reviewing these. Some of these files may be inputs, and these errors must be fixed. Others may be just notifications that files do not exist. If these files are outputs from earlier steps of the mode, these can be ignored (such as loaded networks or skims). You are most likely to encounter this when establishing a new scenario or if you have removed outputs skims or loaded networks from the run folder.

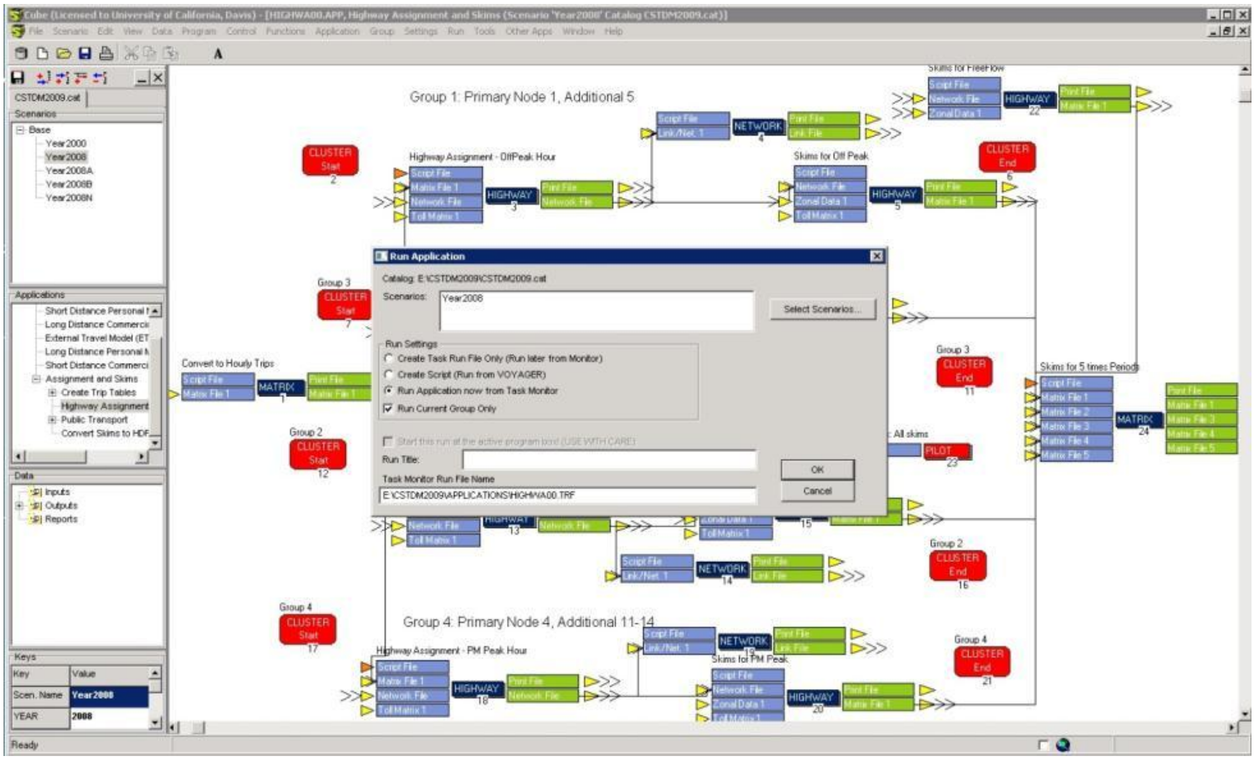


Figure 5. Running a subsection of the model (note the checked box "Run Current Group Only")

To run a subsection of the model, the process is almost identical to running the entire model except that you view the catalog at a group level. For example if you wished to run only the assignment and road network skims, you would select the Assignment and Skims under the application manager (part of the catalog), double click on it so that you could view it. Then make sure that cluster is started and open the Run Application dialog from the Application menu.

Figure 6: Run Application Dialog to Run Current Group Only

To run only the portion of the model you are viewing, check the "Run Current Group Only" check box (Figure 6) before clicking OK and starting the run.

If you wish to run only an individual process, you can simply double click on it in the catalog.

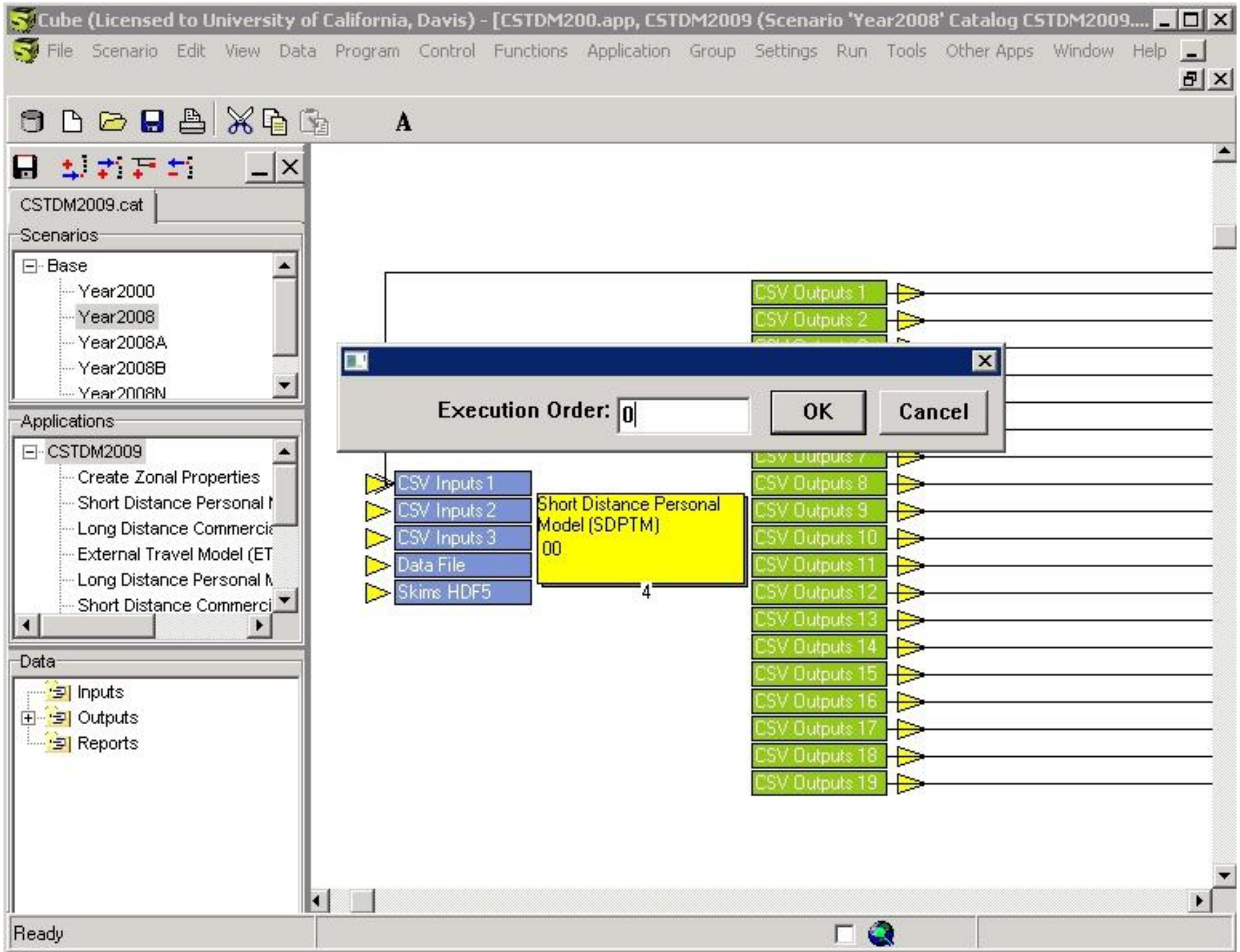


Figure 7. Excluding a step from the run

In a case where you wish to run the majority of the model, but desire to omit one process, you can do so by setting the execution order of that step to zero. Do this by right clicking on the process step you wish to omit, select set execution order and set it to zero (Figure 7). You may receive a notice that the execution order of other components are being adjusted. Running the model will now be the same as for running the full model. Follow a similar procedure for omitting more than one processes, by setting their execution order to zero. Just be careful to document which components are disabled and be sure that all of the outputs are available for any subsequent processes. This method of running the model is most commonly used for debugging, or for using the same demand model outputs for the remainder of the model run (i.e. one or more demand model is disabled so that prior results will be reused).

5. Model Structure

5.1 Narrative

The California Statewide Travel Demand Model (CSTDM) follows the following pattern as it runs.

1. [Step 2 in Figure 8] The initial zonal data is processed from the main zonal properties file into individual zonal properties files for each of the demand models. This reduces the total amount of data that is read in by each model and simplifies the data import process.

2. [Step 2 in Figure 8] The iteration loop begins. If only a single iteration is desired then the loop value can be set to 1, but if multiple iterations are needed the number can be manually controlled.

3. [Step 3 in Figure 8] Short Distance Personal Travel Model (SDPTM): The short distance personal travel model runs as the only computational process while CUBE waits for completion.

4. [Step 4 Figure 8] Short Distance Commercial Vehicle Model (SDCVM): This runs in parallel with the Long Distance Personal Travel Model (LDPTM).[2]

5. [Step 5 Figure 8] Long Distance Personal Travel Model (LDPTM): This runs in parallel with the Short Distance Commercial Vehicle Model (SDCVM).

6. [Step 6 in Figure 8] Long Distance Commercial Vehicle Model (LDCVM): This is a placeholder for a model external to the CSTDM to provide data for freight transportation. This may come from a State Freight Model, or as currently used, a commodity movement dataset exported from the CALSIM (PECAS) model and reprocessed to suit the LDCVM.

7. [Step 7 Figure 8] External Model (ETM): The external gateways model.

8. [Step 8-14 Figure 8] Assignment and Skimming: The outputs from all of the demand models are assembled into a set of Origin-Destination (OD) matrices for assignment. Each of the four time periods (Off Peak, AM Peak, Midday, PM Peak) are assigned, have screen lines counts extracted and then skimmed as simultaneous parallel

[2] Grouping of processes run in parallel was created for optimizing the allocation of computational resources (cores in CUBE Cluster) and minimize the run time of the model.

processes. The skims from each time period are then combined with a set of free-flow skims.

9. [Step 15-18 Figure 8] Short Distance Public Transport covers trips less than 100 miles and is computed by time period. Short distance public transport includes bus and local rail.

10. [Step 19-21 Figure 8] Long Distance Public Transport covers trips greater than 100 miles. Air, conventional rail and access egress skims for long distance transport are calculated.

11. [Step 22 Figure 8] HDF Skim Compilation: All skims are combined into a HDF5 formatted file, and are made available for use as the initial skims by the next iteration of the demand models. The HDF5 format (http://www.hdfgroup.org/HDF5/) is a data model, library and format that focuses on improved I/O (in/out) speed and reliability for large and complex datasets. HDF5 is a free and open source format.

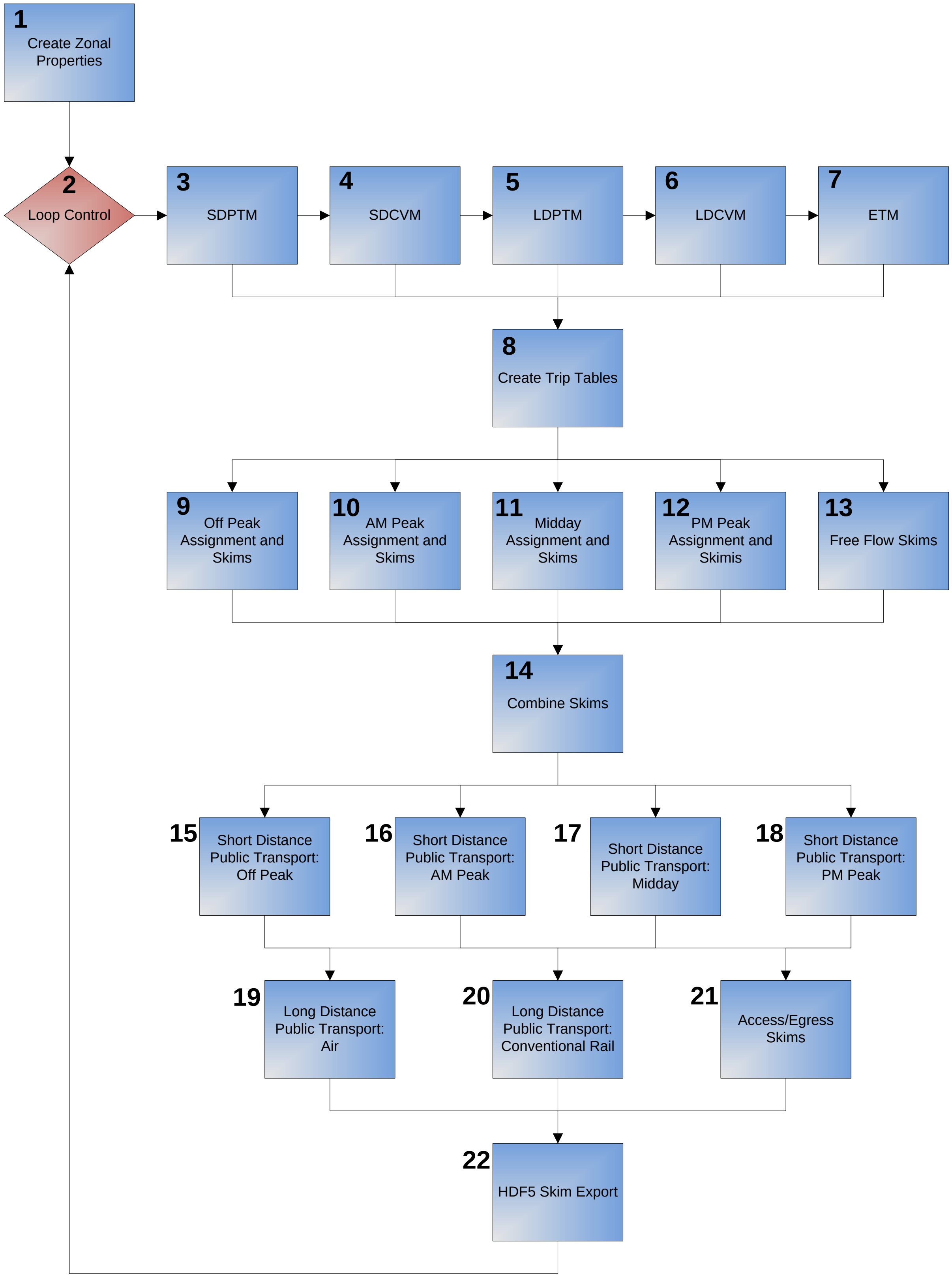


Figure 8: Model Diagram

5.2 Iteration

The CSTDM supports running the model in iterations. By allowing the model to iterate, the mode choice and time period assignments feedback into the demand models to allow a more equitable demand assignment.

To control the number of iterations, right click on the “Loop” control (Process step 2, Figure 8) and change the number of iterations. Each iteration takes the full model run time, so if a single run takes 24 hours, three iterations will take 72, though there may be variations in run times between iterations.

5.3 Cluster

CUBE Cluster is an add-on produced and sold by Citilabs for CUBE. Cluster allows the user to use more than a single CPU for running the model. Without Cluster, the entire model will run on a single CPU and will take several times as long to run. Running CSTDM without Cluster is not recommended as run times for the assignment and skimming portions of the model (not including the demand models) will increase from approximately 10 hours to more than 30.

Each node is a numbered CPU that is waiting for a set of commands to be given to it. In the CSTDM, we start nodes 1 through 15 so that we have 16 CPUs available for our use in CUBE: the initial CPU that started CUBE, and 15 nodes.

To use Cluster in CSTDM, one must first start the nodes before starting the model run. This is done by selecting the tools menu, then Cluster Node Management. The Cluster Node Management panel will open (Figure 9)

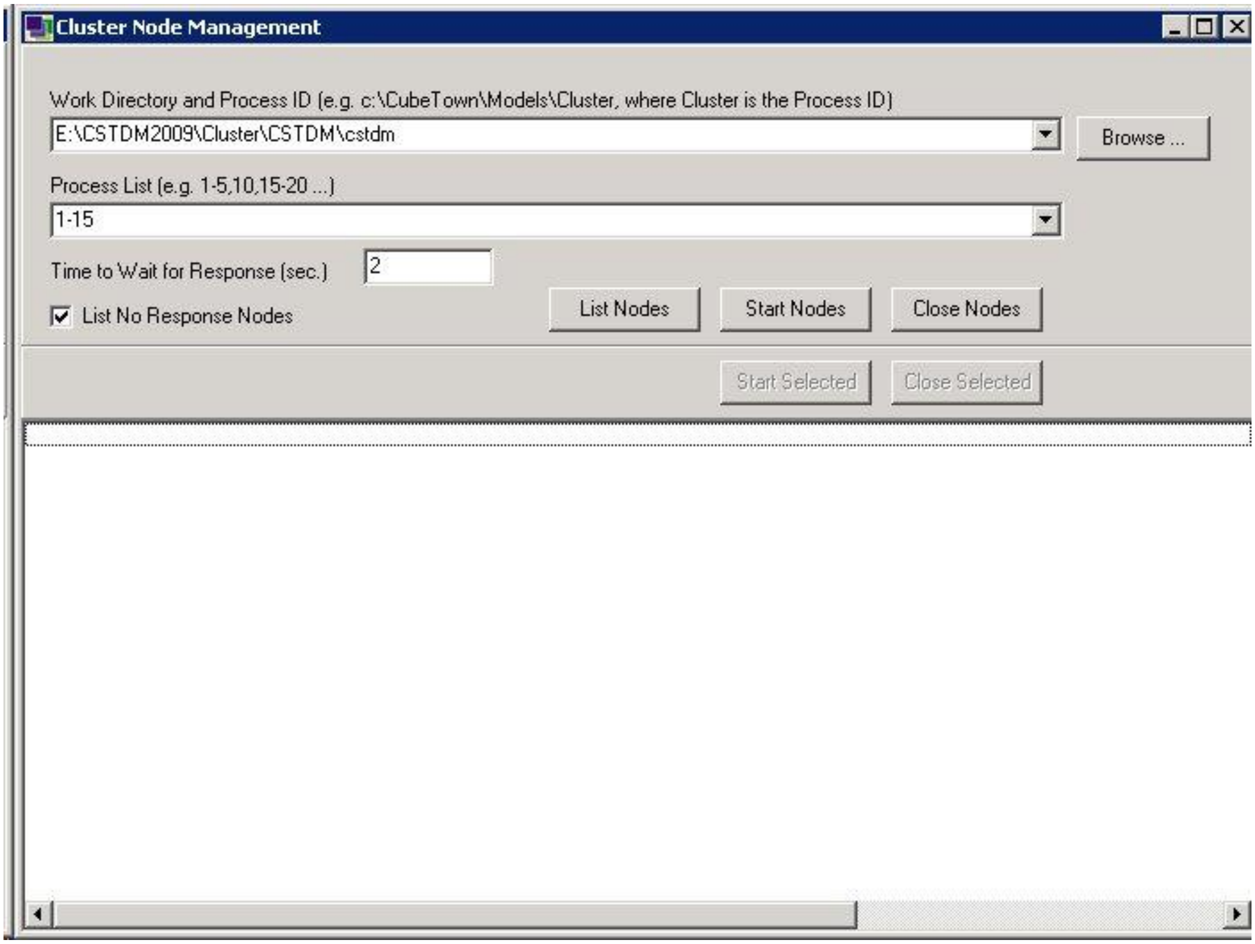


Figure 9: Cluster Node Management

The work directory and process ID must follow the following structure in CSTDM.

“<CSTDM Directory>\Cluster\CSTDM\cstdm”

<CSTDM Directory> is the installation directory for the CSTDM (in our case E:\CSTDM2009). If the installation process has worked correctly there should be a folder called Cluster in this directory, and nested within cluster will be another directory called CSTDM. This is where Cluster will put control scripts for the other nodes.

Cluster has proven to be sensitive to disruptions. In our multi-user, remote desktop accessed platform, we have found that cluster does not respond well to having the remote desktop connection closed and reopened. If possible, we recommend leaving the remote desktop connection active for the duration of the model run.

CUBE Cluster can make use of these nodes in two ways.

1. It can run processes that do not depend on each other in parallel so that the total time taken is the run time required by the longest of the processes. This is called Multistep Distributed Processing, or MDP.
2. It can apply multiple nodes to the same computation to speed up the run time for the process. This is called Intrastep Distributed Processing or IDP. IDP can only be used within the HIGHWAY and MATRIX programs within CUBE.

In some cases these two techniques can be combined as it is done in the highway assignment and skimming component of the model.

5.4 Threads

Threads are very similar to nodes. In fact, each node is actually a thread. Within the context of this document a thread will be used to describe the set of work being done by one CPU. The demand models are written in either JAVA or Python, both of which can make use of multiple threads. Because each of the demand models runs in its own environment after CUBE starts it, they control their own thread usage. Each of these applications has a configuration file that controls how many threads it will use. In this way, we can either allow a demand model to use all available CPUs as we do with the SDPTM, or to throttle it back to use only a subset as we do with the SDCVM and LDPTM, which we then run in parallel.

5.5 User Guide Conventions

The following conventions are used within this user guide. In paths that might change depending on the installation or the scenario being run, text inside carrots (<>) can be thought of as a variable. These are most commonly:

<CSTDM_Directory>, the parent directory containing all of the CSTDM data and code.

<Scenario>, the scenario being run. As setup, there are two scenarios "Year2000" and "Year2008"

<Year>, the year that applies to the scenario, "2000" or "2008"

<time period>, the time period within the day being modeled, most frequently this is "Off Peak", "AM Peak", "Midday", or "PM Peak", but it can vary slightly in some portions of the model that deal only with "Peak" and "Off Peak".

6. Create Zonals

6.1 Description

Figure 10: Create Zonals

The Create Zonals model component takes the base zonal properties file in
<CSTD Directory>/Base/<Scenario>/Controls/Zonal Properties Base <Year>.csv
and subdivides it in to the zonal properties files needed for the SDPTM, SDCVM, LDPTM and ETM models. It is recommended that if you are going to change any of the zonal properties for the CSTDM that you change them in the base zonal properties file. All other zonal properties files (for the individual demand models) will be overwritten during this step with new versions extracted from the base file.

Like all of the components that run in external programming environments, the process works in two steps. The first creates a batch (.bat) file and possibly some control files, and the second executes the batch file. This is illustrated in figure 10.

As part of the process, several additional values are calculated for each TAZ using data from the base zonal properties file and the synthetic population. These include total employment, households, population, income, densities, parking costs, land use totals by type, intrazonal auto speed, and area types.

6.2 Inputs

The input for this procedure is the base zonal properties file with records for each TAZ that include the fields identified in Table 2.

Table 2: Base Zonal Properties Table: Fields and Descriptions

Field Name	Description
TAZ	Transportation Analysis Zone Number
Prim_Sec	Industry: Number of Primary and Secondary Manufacturing workers
Whole	Industry: Number of Wholesale workers
Tran_U	Industry: Number of Transportation and Utilities workers
Office	Industry: Number of Office Worker workers
Retail	Industry: Number of Retail workers
EduMed	Industry: Number of Education or Medical workers
LeisHosp	Industry: Number of Leisure and Hospitality workers
OthServ	Industry: Number of Other Service workers
Military	Industry: Number of Military workers
BluCol	Occupation: Number of Blue Collar workers
Clerical	Occupation: Number of Clerical workers
Education	Occupation: Number of Education workers
Health	Occupation: Number of Health workers
ManBus	Occupation: Number of Management and Business workers
ServNS	Occupation: Number of Non-Sales Service workers
ProfTech	Occupation: Number of Professional and Technical workers
SalesFE	Occupation: Number of Sales, Food and Entertainment workers
E_K8	Education: Number of K-8 grade students

E_912	Education: Number of 9-12 grade students
E_PSE	Education: Number of Post Secondary Education Students
Area_SqMi	TAZ area in square miles
Long	TAZ centroid Longitude
Lat	TAZ centroid Latitude
x-meters	TAZ centroid in California Albers, NAD83
y-meters	TAZ centroid in California Albers, NAD83
Park_Add_Base	Base parking cost for one day
Park_Add_Day	Parking cost for an additional day
County	County Name
FAF_Area	Freight Analysis Framework Area
Calib_5_Name	5 district calibration region name
Calib_22_Name	22 district calibration region name
Calib_5	5 district calibration region number
Calib_22	22 district calibration region number
LUZ	PECAS Land Use Zone
LD_County	County
LD_Region	The region in the 14-region system: 1=AMBAG, 2=Central Coast, 3=Far North, 4=Fresno, 5=Kern, 6=Merced, 7=South San Joaquin, 8=SACOG, 9=SANDAG, 10=San Joaquin, 11=Stanislaus, 12=West Sierra Nevada, 13=MTC, 14=SCAG
LD_Dist	The district in the 25-district system: 1-12=same as region, 13=Alameda, 14=Contra Costa, 15=Marin/Sonoma/Napa, 16=San Francisco, 17=San Mateo, 18=Santa Clara, 19=Solano, 20=Imperial, 21=Los Angeles, 22=Orange, 23=Riverside, 24=San Bernardino, 25=Ventura

6.3 Outputs

The zonal properties for use within CUBE are written to <CSTDM_Directory>/Base/<Scenario>/Controls/Zonal properties_<Year>.csv and .dbf and with the fields indicated in table 3.

Table 3: CUBE Zonal Properties

Field Name	Description
TAZ	Transportation Analysis Zone number
EmpDens	Employee density
PopDens	Population density
Area_SqMi	TAZ area in square miles
AutoSpd	Intrazonal auto speed

The zonal properties for the LDPTM are written to <CSTDM_Directory>\Models\LDPTM\<Scenario>\Inputs\Zonal_Properties_LDPTM.csv with the fields in table 4.

Table 4 LDPTM Zonal Properties

Field Name	Description
TAZ	Transportation Analysis Zone number
LD_County	County name
LD_Region	The region in the 14-region system: 1=AMBAG, 2=Central Coast, 3=Far North, 4=Fresno, 5=Kern, 6=Merced, 7=South San Joaquin, 8=SACOG, 9=SANDAG, 10=San Joaquin, 11=Stanislaus, 12=West Sierra Nevada, 13=MTC, 14=SCAG
LD_Dist	District number
AType	Area Type
LD_Ret	Retail employment
LD_Ser	Service employment
LD_Oth	Other employment

Area_SqMi	TAZ square miles
Pop	Total population
HH	Household count
HHSx_NWy_z	A series of 99 columns, each giving the number of households in particular household segment in the zone. In the column header name, x=household size, y=number of workers, z=1 for low income, 4 for medium income, 7 for high income, plus the number of cars.

The zonal properties for the ETM are written to <CSTDM_Directory>\Models\ETM\<Scenario>\Inputs\Zonal Properties ETM.csv with the fields in table 5.

Table 5: ETM Zonal Properties

Field Name	Description
TAZ	Transportation Analysis Zone number
FAF_Area	Freight Area Framework area
Pop	Population
TotEmp	Employment
CVM_IN	Industrial employment
CVM_RE	Retail employment
CVM_SV	Service employment
CVM_TH	Transportation and Utilities employment
CVM_WH	Wholesale employment

The zonal properties for the SDPTM are written to <CSTDM_Direcotry>\Models\SDPTM\<Scenario>\Inputs\Zonal Properties SDPTM.csv with the fields in table 6.

Table 6: SDPTM Zonal Properties

Field Name	Description
TAZ	Transportation Analysis Zone number
County	County
FAF_Area	Freight Area Framework area
Calib_5_Name	Five district calibration system district name
Calib_22_Name	Twenty two district calibration system district name
Calib_5	Five district calibration system district number
Calib_22	Twenty two district calibration system district number
LUZ	PECAS Land Use Zone
Lat	TAZ Latitude
Long	TAZ Longitude
Area_SqMi	TAZ area
Pop	TAZ population
HH	TAZ households
E_K8	K-8 grade student total
E_912	9-12 grade student total
E_PSE	Post Secondary Education total
TotEmp	Total Employment
Prim_Sec	Industry: Number of Primary and Secondary Manufacturing workers
Whole	Industry: Number of Wholesale workers
Tran_U	Industry: Number of Transportation and Utilities workers
Office	Industry: Number of Office Worker workers
Retail	Industry: Number of Retail workers
EduMed	Industry: Number of Education or Medical workers
LeisHosp	Industry: Number of Leisure and Hospitality workers
OthServ	Industry: Number of Other Service workers
Military	Industry: Number of Military workers

BluCol	Occupation: Number of Blue Collar workers
Clerical	Occupation: Number of Clerical workers
Education	Occupation: Number of Education workers
Health	Occupation: Number of Health workers
ManBus	Occupation: Number of Management and Business workers
ServNS	Occupation: Number of Non-Sales Service workers
ProfTech	Occupation: Number of Professional and Technical workers
SalesFE	Occupation: Number of Sales, Food and Entertainment workers
EmpDens	Employment density
PopDens	Population density
PopEmpDens	Population and Employment density
Park_Base	Base parking cost
Park_Day	Cost to park all day
Park_1H	Cost to park 1 hour
Park_2H	Cost to park 2 hours
Park_3H	Cost to park 3 hours

Zonal properties for the commercial vehicle models are written to <CSTDM_Directory>\models\SDCVM\<Scenario>\Inputs\Zonal Properties SDCVM.csv with the fields in Table 7.

Table 7: SDCVM Zonal Properties

Field Name	**Description**
TAZ	Transportation Analysis Zone
Pop	Population
Income	Income
Area_SqMi	TAZ area in square miles
Lat	Latitude of TAZ centroid
Long	Longitude of TAZ centroid
x-meters	Projected X coordinate of TAZ centroid (California Albers NAD 83)

y-meters	Projected Y coordinate of TAZ centroid (California Albers NAD 83)
EmpDens	Employee density
PopDens	Population density
TotEmp	Total employment
Military	Military employment
CVM_IN	Industrial employment
CVM_RE	Retail employment
CVM_SV	Service Employment
CVM_TH	Transportation and Utilities Employment
CVM_WH	Wholesale Employment
CVM_LU_Type	Land use type
SqrtArea	Square root of the area
CVM_LU_Low	Low density land use flag
CVM_LU_Res	Residential land use flag
CVM_LU_Ret	Retail/commercial land use flag
CVM_LU_Ind	Industrial land use flag
CVM_LU_Emp	Other land use flag
Emp_LU_Lo	Total employment if CVM_LU_Low = 1
Emp_LU_Re	Total employment if CVM_LU_Res = 1
Emp_LU_RC	Total employment if CVM_LU_Ret = 1
Emp_LU_In	Total employment if CVM_LU_In = 1
Emp_LU_EN	Total employment if CVM_LU_Emp = 1

7. SDPTM

7.1 Description

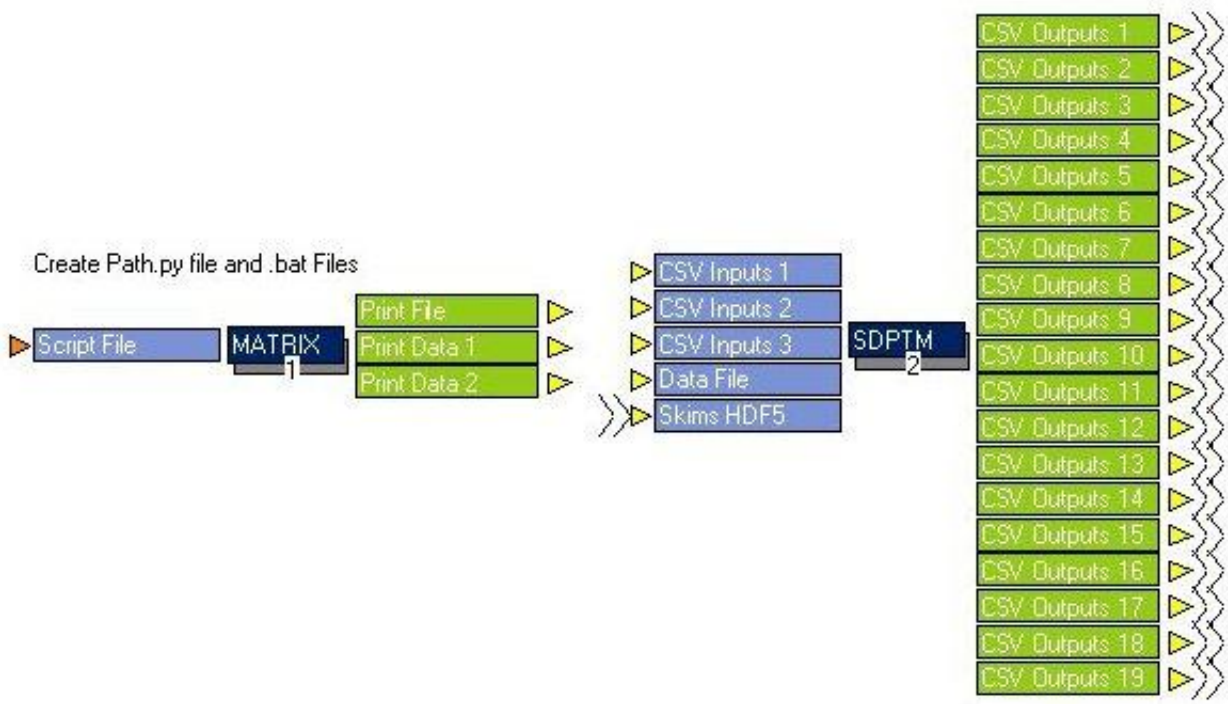


Figure 11: Short Distance Personal Travel Model

The Short Distance Personal Travel Model (SDPTM) executes the python code that produces the output trip lists that will be compiled into the zone to zone volumes. This is a two part process with the first step producing <CSTDM_Directory>\code\sdptm.bat and paths.py and the second executing the .bat file as illustrated in figure 11.

For a complete description of the SDPTM model component, please refer to the three documents that describe the development and estimation of this model.

7.2 Inputs

The SDPTM uses five input files housed in <CSTDM_Directory>\Models\SDPTM\<Scenario>\Inputs\

- Cali.sqlite contains the synthetic population for the year
- Day_Patterns.csv contains the list and proportions of day patterns.
- tazListI.csv contains a list of TAZ, District, latitude and longitude for 19 districts.
- Zonal Properties SDPTM.csv contains all of the zonal properties for each TAZ.
- Skims in a hdf5 format and stored in <CSTDM_Directory>\base\<Scenario>\skims\skims.h5

7.3 Outputs

There are two outputs from the SDPTM:

1. a trip list file (<CSTDM_Directory>\Models\SDPTM\<Scenario>\Outputs\trips_Y.csv where Y is the district) for each of the 19 districts and 4 external regions, containing zone to zone trips by mode and travel for short term destinations (retail, recreation, etc.);
2. a long term destination file for work and school choices (<CSTDM_Directory>\Models\SDPTM\<Scenario>\Outputs\WorkOD_Y.csv where Y is the district).

The fields in the trip table are described in Table 8 and the fields for the long term decision model are in Table 9.

Table 8: Short Term Trip Table Fields

Field Name	**Description**
Model	Model ID
SerialNo	House ID
Person	Person ID
Trip	Trip ID
Tour	Tour ID
HomeZone	Home TAZ

ActorType	Actor Type
OPurp	Purpose key for origin
DPurp	Purpose key for destination
I	Origin zone
J	Destination zone
Time	Time period
Mode	Trip Mode
Leg	Tour leg
TourPurp	Purpose key for tour
Dist	Distance in miles
License	Driver's license
Grade	School grade
TourMode	Tour mode

Table 9: Long Term Trip Table Fields

Fields	**Description**
TAZ	Transportation Analysis Zone
Type	Trip type (Work or School)
Zone	Destination zone
Kind	Type of work or school (School class or occupation)
IncGrp	Income group
Income	Income
Dist	Distance traveled

In addition, specific tables are the results from the auto ownership and driver’s license models that are part of the SDPTM and feed into the final trip generation. Table 10 list fields from the auto ownership model outputs and Table 11 list fields from the driver's license model outputs

Table 10: Auto Ownership Model Outputs

Field Name	Description
Zone	Transportation Analysis Zone
Unique	A unique number for each modeled person
Serial	House ID
Drivers	Number of drivers in the house
Inc	Income class
Veh	Number of vehicles
GQ	General Quarters flag
Persons	Number of persons in the household
Adults	Number of adults in the household

Table 11: Driver's License Model Outputs

Field Name	Description
Zone	Transportation Analysis Zone
Unique	Unique ID
Serial	Household ID
Per	Person number
Type	Person type
Lic	License flag
Age	Person age

8. SDCVM

8.1 Description

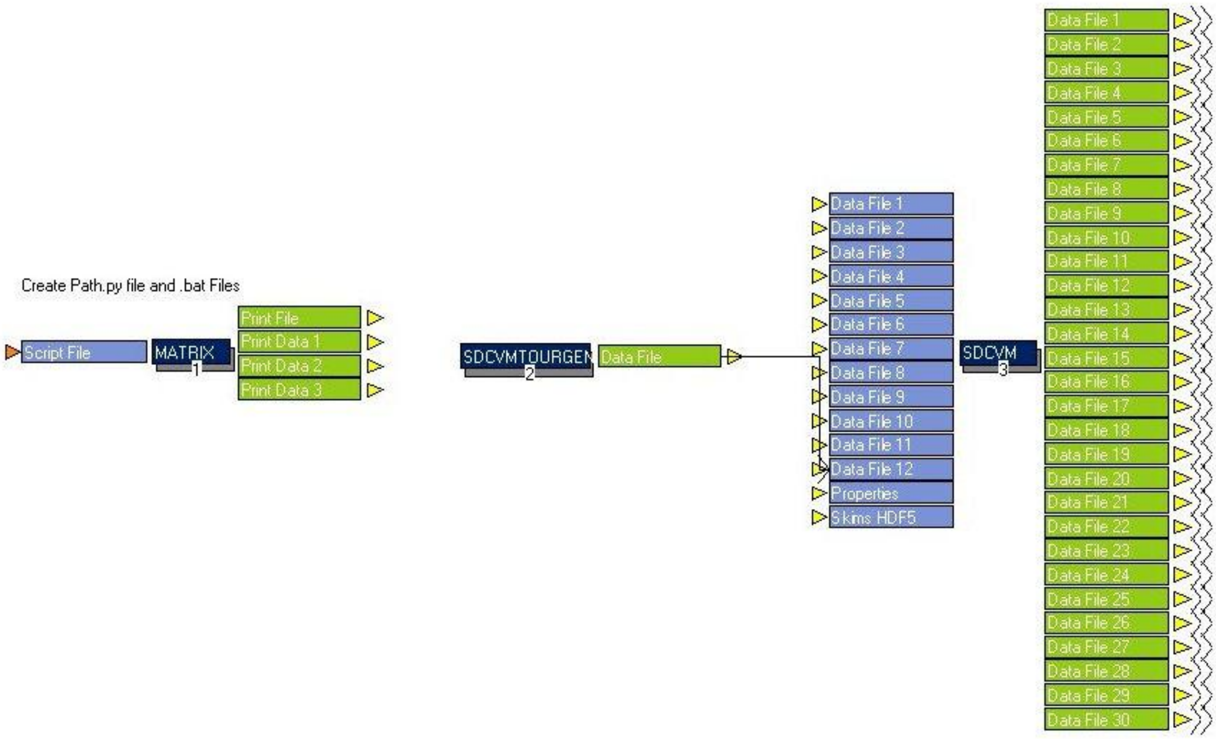


Figure 12: Short Distance Commercial Vehicle Model

The Short Distance Commercial Vehicle Model (SDCVM) operates in three steps. The first generates <CSTDM_Directory>\code\paths.py, sdcvm.bat and sdcvmtourgen.bat. The second executes sdcvmtourgen.bat to produce the tours for the SDCVM, which is executed in the third step as shown in figure 12. The SDCVM is written in Java, and generates short distance (<50mile) commercial vehicle trips.

For a complete description of the SDCVM model component, please refer to the specific document describing the SDCVM model development.

8.2 Inputs

The inputs for the SDCVM include three different forms of table. The first sets the model parameters for each of the five time periods (Offpeak Early [OE.csv], AM Peak [AM.csv], Midday [MD.csv], PM Peak [PM.csv], and Offpeak Late [OL.csv]). The second form specifies the parameters that apply by the industry generating the trip (Fleet Allocator [FA.csv], Primary and Secondary manufacturing [IN.csv], Retail [RE.csv], Service [SV.csv], Transportation and Utilities [TH.csv], Wholesale [WH.csv]). The final input type is Zonal Properties SDCVM.csv generated from the Create Zonal Properties (Section 6).

The SDCVM also uses skims in a hdf5 format and stored in <CSTDM_Directory>\base\<Scenario>\skims\skims.h5

8.3 Outputs

The outputs from the SDCVM are a set of trip tables for each combination of industry and time period. For example, the trip table for AM peak retail trips is:
<CSTDM_Directory>\models\SDCVM\<Scenario>\Outputs\trips_RE_AM.csv
With the fields that are described in table 12.

Table 12: SDCVM Output Structure

Field Name	Description
Model	Model ID
SerialNo	Household ID
Person	Person ID
Trip	Trip ID
Tour	Tour ID
HomeZone	Home TAZ
ActorType	Actor Type
OPurp	Origin Purpose
DPurp	Destination Purpose
I	Trip Origin
J	Trip Destination
Time	Time period
Mode	Model ID
StartTime	Start time
EndTime	End time
StopDuration	Stop duration
TourType	Tour Type
OriginalTimePeriod	Time period for tour start

9. LDPTM

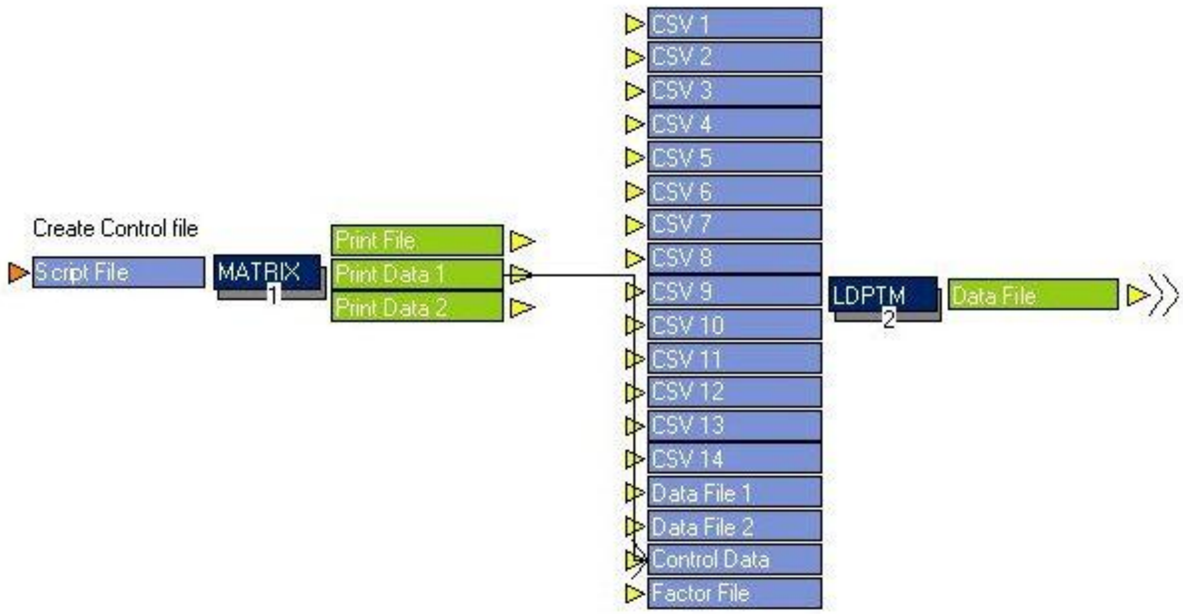


Figure 13: Long Distance Personal Travel Model

9.1 Implementation in CSTDM09 Model Framework

The LDPTM model is implemented using a specially-written Java program. It requires the following inputs:

- **A control file** that specifies the directories and files where the other inputs can be found, the output directory and filenames, and other run parameters. The name of the control file is passed as a command-line argument.
- A **coefficients** file containing all the model coefficients.
- A **zone properties** file in the HDF5 format containing the populations and other properties of the TAZs.
- **Skims** files in the HDF5 format.

This is a two-step process: the first step creates the control file and .bat file, and the second executes it (figure 13).

9.2 Control File

The control file is a simple ASCII text file. Each line of the control file consists of an eight-character key and an arbitrary-length value, separated by any number of spaces. The keys that must be present in the control file, and the meaning of the associated values, are:

- RUNLABEL: the name of the run (e.g. California Long Distance Personal Travel Model: Base Year 2000 Validation).
- RUNDIREC: the directory in which the zone properties file and coefficients file are found.
- PRINTFIL: the name of the print file that any debug output will be printed to.
- COEFFFIL: the name of the coefficients file.
- CARDIREC: the directory in which the car skims are found.
- CARLOSPK: the name of the peak period car skims file.
- CARLOSOP: the name of the off-peak period car skims file.
- AIRDIREC: the directory in which the airport skims are found.
- AIRPORTS: the name of the file containing the airport codes, airport numbers, airport node numbers, and the zones the airports are located in.
- AIRNODPK: the name of the file containing the origin and destination airports used for each zone pair in the peak period.
- AIRNODOP: the name of the file containing the origin and destination airports used for each zone pair in the off-peak period.
- AIRLOSPK: the name of the peak period airport-to-airport skims file.
- AIRLOSOP: the name of the off-peak period airport-to-airport skims file.
- AIRACCPK: the name of the peak period airport access skims file.
- AIRACCOP: the name of the off-peak period airport access skims file.
- AIREGRPK: the name of the peak period airport egress skims file.
- AIREGROP: the name of the off-peak period airport egress skims file.
- CVRDIREC: the directory in which the rail skims are found.
- CVRSTATS: the name of the file containing the rail station numbers, rail station node numbers, and the zones the stations are located in.

- CVRNODPK: the name of the file containing the origin and destination stations used for each zone pair in the peak period.
- CVRNODOP: the name of the file containing the origin and destination stations used for each zone pair in the off-peak period.
- CVRLOSPK: the name of the peak period station-to-station skims file.
- CVRLOSOP: the name of the off-peak period station-to-station skims file.
- CVRACCPK: the name of the peak period station access skims file.
- CVRACCOP: the name of the off-peak period access skims file.
- CVREGRPK: the name of the peak period station egress skims file.
- CVREGROP: the name of the off-peak period station egress skims file.
- ZONEDATA: the name of the zone properties file.
- OUTDIREC: the directory that the outputs will be written to. The program cannot create this directory; i.e. it must exist already in the file structure.
- TRIPLIST: the name of the file that the trip list will be written to.
- FUELCOST: the cost of gasoline in cents per mile (in year $2000 values).
- HSRAVAIL: set as 0 for HSR not available, 1 for HSR available.
- NTHREADS: the number of threads to run.
- TESTORIG and TESTDEST: normally set both as -1. If debug output is needed, each must be a list of zone numbers. The program will print debug output for trips from each origin zone in TESTORIG to the corresponding destination zone in TESTDEST. For example, TESTORIG 100, 101, 102 and TESTDEST 305, 306, 307 will produce debug output for zone 100 to zone 305, zone 101 to zone 306, and zone 102 to zone 307.

If any of these keys are missing, the program will halt and report an error.

Notes:

- **If HSRAVAIL is set to 1**, the control file must provide the location of HSR skims in addition to the car, air, and rail skims, using the keys HSRDIREC, HSRSTATS, HSRNODPK, HSRNODOP, HSRLOSPK, HSRLOSOP, HSRACCPK, HSRACCOP, HSREGRPK, and HSREGROP. These keys work the same way as the similar keys for conventional rail.

- The directory names must end in a backslash (e.g. “C:\LDPTM\Auto\” not “C:\LDPTM\Auto”).
- The directory names and the run label can have spaces in them. The filenames must not contain spaces.

9.3 Coefficients File

The coefficients file is written in the output format used by ALOGIT, as that was the source of most of the coefficients. Tables 13 through 16 show which coefficient number in the ALOGIT file corresponds to each coefficient in the specifications.

The trip frequency models use coefficients from ALOGIT file models 22, 23, 24, and 25 for business, commute, recreation, and other, respectively.

Table 13: Coefficients File Numbers – Trip Frequency Models

Parameter	Coefficient Number	Coefficient Name
Intra-Region Accessibility	1	Regacc
Logsum Destination Choice	3	Llogsum
1-Person HH	5	Onephh
3+ Person HH	6	threephh
Ratio Workers / Size	12	Wkrspps
No Workers in HH	16	Nowkrs
Medium Income	7	Medinc
High Income	8	Highinc
Cars < Workers	11	Carsltw
SACOG Resident	13	Sacog
SANDAG Resident	14	Sandag
MTC Resident	15	Mtc
SCAG Resident	17	Scag
1 Trip	21	const1
2 Trips	22	const2

The destination choice models use coefficients from model 16 for business/commute and model 17 for recreation/other.

Table 14: Coefficients File Numbers – Destination Choice Models

Parameter	Coefficient Number	Coefficient Name
Mode Choice Logsum	1	mlogsum
Distance (Miles)	2	Distance
Distance Squared	3	Distsqu
Distance Cubed	4	Distcub
Urban Destination	5	Durban
Rural Destination	6	Drural
Urban to Urban	7	Urburb
Suburban to Suburban	8	Subsub
Rural to Rural	9	Rurrur
AMBAG	41	AMBAG
Central Coast	42	CC
Far North	43	FN
Fresno	44	FM
Kern	45	Kern
Merced	46	Merced
S. San Joaquin	47	SSJ
SACOG	48	SACOG
SANDAG	49	SANDAG
San Joaquin	50	SJ
Stanislaus	51	Stan
W. Sierra Nevada	52	WSN
Alameda	53	MTC
Contra Costa	54	MTC
Marin/Sonoma/Napa	55	MTC
San Francisco	56	MTC

San Mateo	57	MTC
Santa Clara	58	MTC
Solano	59	MTC
Los Angeles	61	SCAG
Orange	62	SCAG
Riverside	63	SCAG
San Bernardino	64	SCAG
Ventura	65	SCAG
Imperial	60	SCAG
MTC to SCAG	71	Mtcscag
SCAG to MTC	75	Scagmtc
MTC to SANDAG	72	mtcsandag
SANDAG to MTC	77	sandagmtc
MTC to SACOG	79	mtcsacog
SACOG to MTC	80	sacogmtc
SCAG to SANDAG	81	scagsandag
SANDAG to SCAG	82	sandagscag
SCAG to SACOG	76	scagsacog
SACOG to SCAG	73	sacogscag
SANDAG to SACOG	78	sandagsac
SACOG to SANDAG	74	sacogsand
Retail Emp – Low Income	101	Loincret
Retail Emp – Medium Income	103	mdincret
Retail Emp – High Income	105	Hiincret
Service Emp – Low Income	102	Loincsvc
Service Emp – Medium Income	104	mdincsvc
Service Emp – High Income	106	Hiincsvc

The main mode choice models use coefficients from model 11 for business/commute and model 12 for recreation/other.

Table 15: Coefficients File Numbers – Main Mode Choice Models

Parameter	Coefficient Number	Coefficient Name
Money Cost (Dollars)	1	Cost
In-Vehicle Time (Minutes)	2	Time
Headway (Minutes)	4	Freq
Reliability (% on time)	3	Reli
Access Mode Logsum	5	Accls
Egress Mode Logsum	6	Egrls
Travel in a Group – Car	104	c-group
Travel in a Group – Air	210	a-group
Household Size (max. 3) – Car	107	c-hhsize
Fewer than 2 Cars (HH size > 1 Person) – Car	106	c-carslt2
High Income – Air	208	a-hiinc
High Income – Conventional Rail	408	r-hiinc
High Income – High-Speed Rail	308	h-hiinc
Air Constant	200	a-const
Conventional Rail Constant	400	r-const
High-Speed Rail Constant	300	h-const
LAX to SFO	211	{lax-sfo}
SFO to LAX	212	{sfo-lax}
LAX to OAK	213	{lax-oak}
OAK to LAX	214	{oak-lax}
LAX to SJC	215	{lax-sjc}
SJC to LAX	216	{sjc-lax}
LAX to SAC	217	{lax-sac}
SAC to LAX	218	{sac-lax}
BUR to SFO	221	{bur-sfo}
SFO to BUR	222	{sfo-bur}
BUR to OAK	223	{bur-oak}

OAK to BUR	224	{oak-bur}
BUR to SJC	225	{bur-sjc}
SJC to BUR	226	{sjc-bur}
BUR to SAC	227	{bur-sac}
SAC to BUR	228	{sac-bur}
ONT to SFO	231	{ont-sfo}
SFO to ONT	232	{sfo-ont}
ONT to OAK	233	{ont-oak}
OAK to ONT	234	{oak-ont}
ONT to SJC	235	{ont-sjc}
SJC to ONT	236	{sjc-ont}
ONT to SAC	237	{ont-sac}
SAC to ONT	238	{sac-ont}
SNA to SFO	241	{sna-sfo}
SFO to SNA	242	{sfo-sna}
SNA to OAK	243	{sna-oak}
OAK to SNA	244	{oak-sna}
SNA to SJC	245	{sna-sjc}
SJC to SNA	246	{sjc-sna}
SNA to SAC	247	{sna-sac}
SAC to SNA	248	{sac-sna}
SAN to SFO	251	{san-sfo}
SFO to SAN	252	{sfo-san}
SAN to OAK	253	{san-oak}
OAK to SAN	254	{oak-san}
SAN to SJC	255	{san-sjc}
SJC to SAN	256	{sjc-san}
SAN to SAC	257	{san-sac}
SAC to SAN	258	{sac-san}

The access and egress mode choice models use coefficients from models 1, 2, 3, and 4 for business/commute access, business/commute egress, recreation/other access, and recreation/other egress, respectively.

Table 16: Coefficients File Numbers – Access/Egress Mode Choice Models

Parameter	Coefficient Number	Coefficient Name
Money Cost (Dollars)	2	Cost
In-Vehicle Time (Minutes)	1	Ivt
In-Vehicle Time (Minutes) for Pick Up/Drop Off	4	aivt-pkup
Auto Distance (Miles)	5	adis-taxi
Out of Vehicle Time	12	Ovt
Airport is LAX – Drive and Park	111	dp-laxacc
Airport is SFO – Drive and Park	112	dp-sfoacc
Airport is SAN – Drive and Park	115	dp-sanacc
Airport is SJC – Drive and Park	114	dp-sjcacc
Main Mode is Air – Walk	603	wk-air
Main Mode is Conventional Rail – Drive and Park	103	dp-cvr
Main Mode is Conventional Rail – Rental Car	203	rc-cvr
Main Mode is Conventional Rail – Taxi	403	tx-cvr
Main Mode is Conventional Rail – Transit	503	tr-cvr
Main Mode is High-Speed Rail – Drive and Park	104	dp-hsr
Main Mode is High-Speed Rail – Rental Car	204	rc-hsr
Main Mode is High-Speed Rail –	404	tx-hsr

Taxi		
Main Mode is High-Speed Rail – Transit	504	tr-hsr
Travel Alone – Drive and Park	105	dp-alone
Travel Alone – Rental Car	205	rc-alone
Travel Alone – Taxi	405	tx-alone
Travel Alone – Transit	505	tr-alone
Household Size – Pick Up/Drop Off	302	sp-hhsize
No Cars in HH – Rental Car	206	rc-nocars
No Cars in HH – Transit	506	tr-nocars
Fewer than 2 Cars in HH (HH size > 1 Person) – Drive and Park	107	dp-carsltw
Fewer than 2 Cars in HH (HH size > 1 Person) – Transit	507	tr-carsltw
Low Income – Drive and Park	108	dp-lowinc
Low Income – Rental Car	208	rc-lowinc
Low Income – Taxi	408	tx-lowinc
Low Income – Transit	508	tr-lowinc
High Income – Drive and Park	109	dp-hiinc
High Income – Rental Car	209	rc-hiinc
High Income – Taxi	409	tx-hiinc
High Income – Transit	509	tr-hiinc
Drive and Park Access Constant	101	dp-acc
Drive and Park Egress Constant	102	dp-egr
Rental Car Access Constant	201	rc-acc
Rental Car Egress Constant	202	rc-egr
Taxi Access Constant	401	tx-acc
Taxi Egress Constant	402	tx-egr
Transit Access Constant	501	tr-acc

Transit Egress Constant	502	tr-egr
Walk Access Constant	601	wk-acc
Walk Egress Constant	602	wk-egr

9.4 Zonal Properties File

The Zonal Properties file is a .csv file with a record for each TAZ in the zone system. Table 17 summarizes the fields that should appear in each record in the order given.

Table 17: Fields in the Zonal Properties File

Column Header	Field Description
FINAL_TA	The TAZ number
COUNTY	The county
REGION	The region in the 14-region system: 1=AMBAG, 2=Central Coast, 3=Far North, 4=Fresno, 5=Kern, 6=Merced, 7=South San Joaquin, 8=SACOG, 9=SANDAG, 10=San Joaquin, 11=Stanislaus, 12=West Sierra Nevada, 13=MTC, 14=SCAG
DISTRICT	The district in the 25-district system: 1-12=same as region, 13=Alameda, 14=Contra Costa, 15=Marin/Sonoma/Napa, 16=San Francisco, 17=San Mateo, 18=Santa Clara, 19=Solano, 20=Imperial, 21=Los Angeles, 22=Orange, 23=Riverside, 24=San Bernardino, 25=Ventura
ATYPE	The density of the zone: 1=Urban Core, 2=Urban, 3=High Suburban, 4=Low Suburban, 5=Rural
RET	The total number of retail jobs in the zone
SER	The total number of service jobs in the zone
OTH	The total number of other jobs in the zone
SQAREMIL	The area of the zone in square miles
TOT_POP	The total population in the zone
TOT_HH	The total number of households in the zone
HHSx_NWy_z	A series of 99 columns, each giving the number of

	households in particular household segment in the zone. In the column header name, x=household size, y=number of workers, z=1 for low income, 4 for medium income, 7 for high income, plus the number of cars.

9.5 Skims Files

The skims files are all HDF5 files with various record structures. They are divided into 5 types: car skims, station correspondence files, passenger mode stops files, station-to-station skims, and access/egress skims.

The car skims provide the level of service by car between every pair of zones. There is a car skims file for each of the time periods. Table 18 shows the fields that should appear in each record in the order given.

Table 18: Fields in the Car Skims Files

Column Header	Field Description
I	The TAZ number of the origin zone
J	The TAZ number of the destination zone
TIME	The travel time by car between the zones, in minutes
TOLL	The total road toll between the zones, in dollars
DIST	The road distance between the zones, in miles

Each airport and rail station has both an internal station number (usually the numbers from 1 to the number of airports/stations) and a node number used in the network encoding. The station correspondence files map these two representations to each other, and tell which zone each airport/station is in. Table 19 shows the fields that should appear in each record in the order given.

Table 19: Fields in the Station Correspondence Files

Column Header		**Field Description**
Airport File	**Station File**	
AIRPORT	RAIL_STA	The 3-letter airport code (left blank for rail stations)
NUMBER	RAIL_STA_N	The internal station number
NODE	RAIL_NOD	The node number
TAZ	NEW_TAZ	The TAZ that the airport/station is located in

The passenger mode stops files tell which airports/stations will be used for trips between each pair of zones for which that mode is available. There is an airport stops file and a rail station stops file for both the peak period and the off-peak period. The fields are listed in Table 20.

Table 20: Fields in the Passenger Mode Stops Files

Column Header	**Field Description**
I	The TAZ number of the origin zone
J	The TAZ number of the destination zone
ACC_STATION	The node number of the airport/station where the passenger transfers from the access mode to the main mode (the *origin* airport/station)
EGR_STATION	The node number of the airport/station where the passenger transfers from the main mode to the egress mode (the destination airport/station)

The station-to-station skims files provide the level of service by air or rail between each pair of airports/stations. There is an airport-to-airport file and a rail station-to-station file for both the peak period and the off-peak period. The fields are listed in Table 21.

Table 21: Fields in the Station-to-Station Skims Files

Column Header	Field Description
I	The internal station number (NOT the node number) for the origin airport/station
J	The internal station number for the destination airport/station
FARE	The fare, in dollars, to ride the passenger mode between the two airports/stations
IVTIMES	The travel time in minutes between the two airports/stations – zero indicates no route available
HEADWAY	The average time, in minutes, between departures from the origin airport/station to the destination airport/station
RELIABILITY	The percent reliability of trips from the origin airport/station to the destination airport/station, as defined in the specifications

The access/egress skims files provide the level of service for access and egress trips, both by car and by transit. Both access and egress skims are listed as zone-to-zone skims; they refer to the zone containing the airports/stations rather than the airports/stations themselves. There is an airport access/egress file and a rail station access/egress file for both the peak period and the off-peak period. The fields are listed in Table 22.

Table 22: Fields in the Access/Egress Skims Files

Column Header	Field Description
I	The origin zone number for the access/egress trip. For access trips, this is the origin zone for the overall trip, while for egress trips, this is the zone containing the destination airport/station.
J	The destination zone number for the access/egress trip. For access trips, this is the zone containing the origin airport/station, while for egress trips, this is the destination zone for the overall trip.
TRA_FARE	The fare, in dollars, to ride local transit between the two zones
TRA_IVT	The in-vehicle time in minutes for the transit trip between the two zones – zero indicates transit not available
TRA_OVT	The time in minutes spent waiting for transit or walking to or from transit in the course of the transit trip between the two zones
AUTO_DIST	The distance by car between the zones, in miles
AUTO_TIME	The travel time by car between the zones, in minutes
AUTO_TOLL	The total road toll between the zones, in dollars

9.6 Output

The output for the LDPTM consists of a trip list in the HDF5 format; each record is a single trip. The fields used for the trip list are consistent with the other components of the CSTDM. Table 23 lists the fields and explains how they are adapted to the LDPTM.

Table 23: Fields in the LDPTM Output Format

Column Header	Field Description
Model	Always 2 for the LDPTM
SerialNo	A unique number assigned to each trip, starting at 1 and incrementing
Person	Always 1 for the LDPTM
Trip	Always 1 for the LDPTM
Tour	Always 1 for the LDPTM
HomeZone	The origin zone for the trip
ActorType	Always “Person” for the LDPTM
OPurp	The purpose of the trip: “Bus” for business, “Com” for commute, “Rec” for recreation, “OtL” for other
DPurp	Same as OPurp for the LDPTM
I	The origin zone for the trip if the direction is “from home”, otherwise the destination zone for the trip
J	The destination zone for the trip if the direction is “from home”, otherwise the origin zone for the trip
Time	The time period that the trip occurs in: 2 for AM peak, 3 for midday, 4 for PM peak, 5 for late off-peak
Mode	The main mode, with occupancy indicated for car trips: “SOV”, “HOV2”, or “HOV3” for car trips of the corresponding occupancy, “Air” for air, “Rail” for rail, “HSR” for high-speed rail
AccMode	“Park” for Drive and Park, “Rent” for Rental Car, “Drop” for Drop Off, “Taxi” for Taxi, “Trans” for Transit, “Walk” for Walk

EgrMode	"Park" for Drive and Park, "Rent" for Rental Car, "Drop" for Pick Up, "Taxi" for Taxi, "Trans" for Transit, "Walk" for Walk
HHSize	Household size (1, 2, 3, or 4; 4 indicates 4 or more)
HHWks	Number of workers in household (0, 1, or 2; 2 indicates 2 or more)
HHInc	"Low" for low income, "Med" for medium income, "High" for high income
HHCars	Number of cars in the household (0, 1, or 2; 2 indicates 2 or more)
OSNode	The origin airport/station Cube node number for the trip by air/rail
OStation	The origin airport/station internal reference number
DSNode	The destination airport/station Cube node number for the trip by air/rail
DStation	The destination airport/station internal reference number

Note: The Access and Egress modes given in the output are those used for the original outbound trip from home to the airport / rail station, even when the trip record contains data on the return trip back to home.

The car trips for each car mode and time period in the above trip list are combined with car trips from the other model components of the CSTDM, and assigned to the road networks for each time period.

The current implementation of the LDPTM does not explicitly extract (and provide for assignment) the car mode components of the access and egress choice for air and rail trips.

10. LDCVM

10.1 Description

The Long Distance Commercial Vehicle Model imports a set of origin-destination tables that originate from the CALSIM (PECAS) in this case. In the future other viable sources, e.g. a dedicated Freight Transportation Model for California, may exist to supplement or replace PECAS in this role.

A complete description of the LDCVM model component is contained in the LDCVM document, which provides detailed information on the development of this model component.

10.2 Outputs

As setup for delivery of the model, the LDCVM does no include processing. It essentially provides an external table of trips with the fields in table 24 to the Assignment and Skimming process.

Table 24: LDCVM Input File Description

Field Name	Description
origin	Origin TAZ
destination	Destination TAZ
type	Load Type

11. ETM

11.1 Description

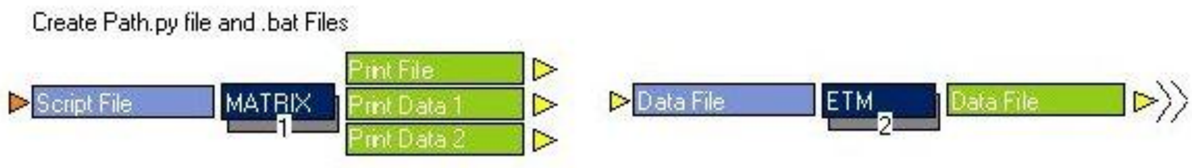


Figure 14: External Travel Model

The External Travel Model operates in two phases as shown in figure 14. The first produces <CSTDM_Directory>\code\etm.bat and path.py. The second executes the batch file to run the External Travel Model which is written in Java.

A complete description of the ETM model component can be found in the ETM document, which provides detailed information on the development of this model component.

11.2 Inputs

Inputs to the ETM consist of two tables and network skims:

- Externals.csv with targets for the External Travel Model
- Zonal Properties ETM.csv with zonal properties
- Skims in a hdf5 format and stored in <CSTDM_Directory>\base\<Scenario>\skims\skims.h5

Externals.csv contains the fields as described in Table 25. The Zonal Properties for the ETM model component are in Table 26.

Table 25: Fields in Externals.csv

Field Name	Description
TAZ	Transportation Analysis Zone

ExtDist	External District
Name	Name of external district
Volume	Total Volume
E-I	Proportion of volume that is external to internal
I-E	Proportion of volume that is internal to external
E-E	Proportion of volume that is external to external
CarLocal	Proportion of volume that is local car trips
CarLong	Proportion of volume that is long distance car trips
Medium	Proportion of volume that is medium trucks
Heavy	Proportion of volume that is heavy trucks
Time_1_IE	Proportion of volume that is in off peak early as internal to external
Time_2_IE	Proportion of volume that is in am peak as internal to external
Time_3_IE	Proportion of volume that is in midday as internal to external
Time_4_IE	Proportion of volume that is in pm peak as internal to external
Time_5_IE	Proportion of volume that is in off peak late as internal to external
Time_1_EI	Proportion of volume that is in off peak early as external to internal
Time_2_EI	Proportion of volume that is in am peak as external to internal
Time_3_EI	Proportion of volume that is in midday as external to internal
Time_4_EI	Proportion of volume that is in pm peak as external to internal
Time_5_EI	Proportion of volume that is in off peak late as external to internal
EE_4	Proportion of external to external trips to external zone 4
EE_14	Proportion of external to external trips to external zone 14
EE_16	Proportion of external to external trips to external zone 16
EE_31	Proportion of external to external trips to external zone 31
EE_35	Proportion of external to external trips to external zone 35
EE_39	Proportion of external to external trips to external zone 39

EE_42	Proportion of external to external trips to external zone 42
EE_43	Proportion of external to external trips to external zone 43
EE_44	Proportion of external to external trips to external zone 44
EE_45	Proportion of external to external trips to external zone 45
EE_46	Proportion of external to external trips to external zone 46
EE_47	Proportion of external to external trips to external zone 47
EE_48	Proportion of external to external trips to external zone 48
EE_49	Proportion of external to external trips to external zone 49
EE_50	Proportion of external to external trips to external zone 50
EE_51	Proportion of external to external trips to external zone 51

Table 26: Fields in Zonal Properties ETM.csv

Field Name	**Description**
TAZ	Transportation Analysis Zone number
FAF_Area	Freight Area Framework area
Pop	Population
TotEmp	Employment
CVM_IN	Industrial employment
CVM_RE	Retail employment
CVM_SV	Service employment
CVM_TH	Transportation and Utilities employment
CVM_WH	Wholesale employment

11.3 Outputs

The outputs from the ETM are a table of trips with the fields described in Table 27.

Table 27: ETM Output Fields

Field Name	Description
Model	Model ID
SerialNo	Household ID
Person	Person ID
Trip	Trip ID
Tour	Tour ID
HomeZone	Home TAZ
ActorType	Actor type
OPurp	Origin purpose
DPurp	Destination purpose
I	Origin TAZ
J	Destination TAZ
Time	Time period
Mode	Mode
Ext	External zone name
Int	Internal region name

12. Assignment and Skimming

12.1 Description

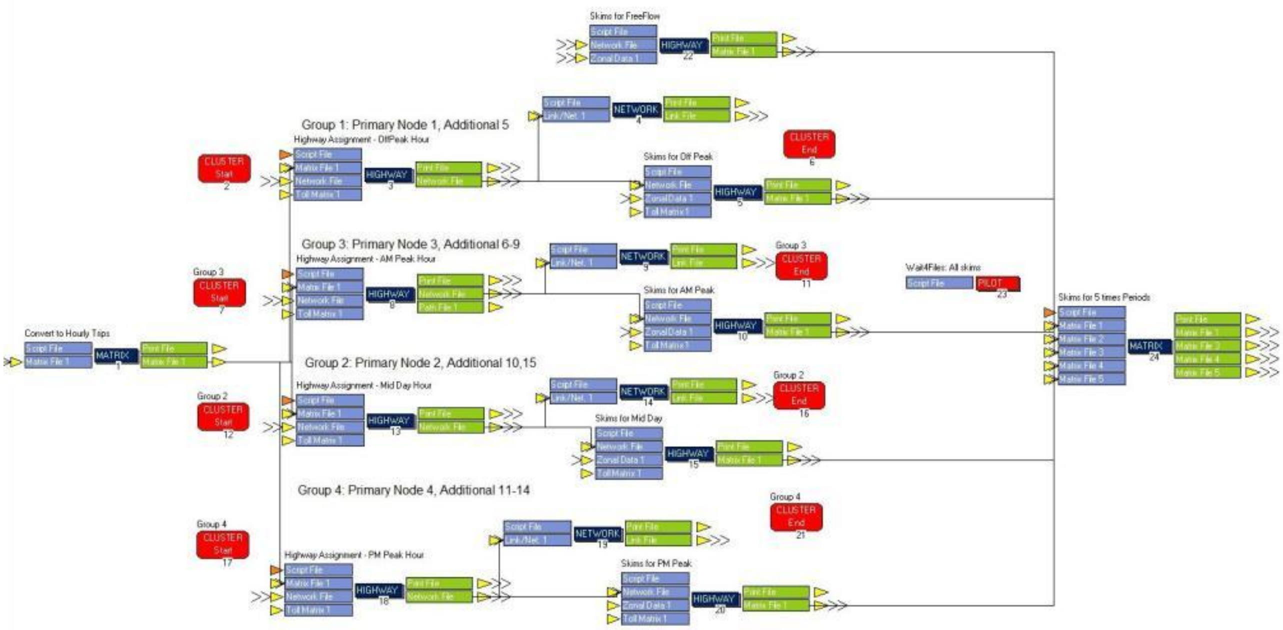


Figure 15: Assignment and Skimming

The Assignment and highway skimming portion of the CSTDM has three primary segments as shown in figure 15. The first converts trip tables into OD tables for assignment by time period.

The second section assigns those trips to the highway network by each time period, extracts screenline data and produces skims for the time period. A set of freeflow skims is also generated. The last section compiles all of the skims to a single dataset.

In this portion of the model, each of the assignment and skimming processes by time period is run in parallel on a separate set of nodes. The number of nodes has been assigned to try to equalize the run times for each of the time periods.

12.2 Inputs

Inputs to the trip table conversion are the trip tables from each of the five demand models. These are combined to create a matrix that will be loaded onto the networks by time period and mode.

The highway assignment step requires a network for each time period and a toll matrix that includes the tolls for each tolled road link. The attributes of each of these files are documented in the Network documentation.

Highway networks are located in
<CSTDM_Directory>\base\<Scenario>\HwyNetworks\AlphaNetwork_<time period>_<year>.net
and have the fields described in table 28.

Table 28: Fields in the Network Files

For details please see the network documentation.

Field Name	Description
A	A node for 1-way link
B	B node for 1-way link
DISTANCE	Link length (miles)
LANES	Number of lanes (1 direction)
SPEED	Free flow speed (mph)
CAPACITY	Vehicles per lane per hour
FTYPE	Facility type
AREATYPE	Area type
AREATYPENUM	Area type number
USE	Vehicles allowed
TOLL	Flag for link with toll
TIME_INIT	Initial travel times (minutes)
FACILITY	Type of link (e.g. Walk Links)

NAME	Street Name
EDITED	Edit status
ONRAMP	Flag for link with toll
OFFRAMP	Flag for link with toll
SCREENLINE	Flag for identifying links in screenlines

Toll matrices are in <CSTDM_Directory>\base\<Scenario>\HwyNetworks\tolls_<time period>.dbf and have the structure described in table 29.

Table 29: Toll File Structure

Field Name	**Description**
ON_RAMP	Toll ID
OFF_RAMP	Toll ID
TOLL	Toll ID
COST_SOV	Cost for a SOV
COST_HOV2	Cost for a HOV2
COST_HOV3	Cost for a HOV3
COST_TRUCK	Cost for a Truck

12.3 Outputs

Outputs include the Loaded networks for each time period as <CSTDM_Directory>\base\<Scenario>\LoadedNetworks\ HwyNetwork_Loaded_<time period>_<year>.NET.

The attributes contained in the loaded network include all attributes from Table 28 and add the following attributes in Table 30.

Table 30: Loaded Network Fields in Addition to Those from the Unloaded Network

Field Name	Description
V_2	Total volume
Time_2	Travel time on link
VC_2	Volume/capacity ratio
CSPD_2	Congested speed
VDT_2	Vehicle-distance traveled
VHT_2	Vehicle-hours traveled
V1_2	Volume, SOV, short distance
V2_2	Volume, HOV2, short distance
V3_2	Volume, HOV3, short distance
V4_2	Volume, Light truck, short distance
V5_2	Volume, Medium truck, short distance
V6_2	Volume, Heavy truck, short distance
V7_2	Volume, SOV, long distance
V8_2	Volume, HOV2, long distance
V9_2	Volume, HOV3, long distance
V10_2	Volume, Heavy truck, long distance
V11_2	Volume, SOV, external trips
V12_2	Volume, HOV2, external trips
V13_2	Volume, HOV3, external trips
V14_2	Volume, Medium truck, external trips
V15_2	Volume, Heavy truck, external trips
V1T_2	The volume in the opposite direction
V2T_2	The volume in the opposite direction
V3T_2	The volume in the opposite direction
V4T_2	The volume in the opposite direction
V5T_2	The volume in the opposite direction
V6T_2	The volume in the opposite direction

V7T_2	The volume in the opposite direction
V8T_2	The volume in the opposite direction
V9T_2	The volume in the opposite direction
V10T_2	The volume in the opposite direction
V11T_2	The volume in the opposite direction
V12T_2	The volume in the opposite direction
V13T_2	The volume in the opposite direction
V14T_2	The volume in the opposite direction
V15T_2	The volume in the opposite direction

Other outputs include:

- Skims for each time period by mode
  <CSTDM_Directory>\base\<Scenario>\skims\Auto
- A combined set of skims for auto travel
  <CSTDM_Directory>\base\<Scenario>\skims\Auto
- Screenline comparisons,
  <CSTDM_Directory>\base\<Scenario>\screenlines\<time period>_<year>.dbf.

A description of the fields used for the screenline comparison is found in table 31.

Table 31: Screenline Comparison Fields

Field Name	Description
SCREENLINE	Screenline ID
STREET_NAM	Street name
SCLN_LOCAT	Screenline location
COUNTY_1	County
FACILITY	Facility description
LANES	Number of Lanes
SPEED	Speed limit
A	A node
B	B node

V1_2	Volume, SOV, short distance
V2_2	Volume, HOV2, short distance
V3_2	Volume, HOV3, short distance
V4_2	Volume, Light truck, short distance
V5_2	Volume, Medium truck, short distance
V6_2	Volume, Heavy truck, short distance
V7_2	Volume, SOV, long distance
V8_2	Volume, HOV2, long distance
V9_2	Volume, HOV3, long distance
V10_2	Volume, Heavy truck, long distance
V11_2	Volume, SOV, external trips
V12_2	Volume, HOV2, external trips
V13_2	Volume, HOV3, external trips
V14_2	Volume, Medium truck, external trips
V15_2	Volume, Heavy truck, external trips
SOV_<time>	Total SOV in the Time period
HOV2_<time>	Total HOV2 in the Time period
HOV3_<time>	Total HOV3 in the time period
LCV_<time>	Total light commercial vehicles in the time period
MCV_<time>	Total medium commercial vehicles in the time period
HCV_<time>	Total heavy commercial vehicles in the time period
AUTO_<time>	Total autos in the time period
TRUCK_<time>	Total trucks in the time period
TOT_<time>	Total volume in the time period
CAPACITY	Capacity
V_2_<time>	Volume
VC_2_<time>	Volume/capacity ratio

12.4 Travel Skims Extracted for CSTDM2009

The following sections describe the zone to zone travel skims extracted from the CUBE network descriptions and assignment results for use in the different sub-models of the CSTDM.

12.5 Travel Skims Extracted for the SDPTM

In total 76 zone origin to destination travel skims are extracted for this sub-model, by mode, time period and skim property. Table 32 summarizes these skims.

Table 32: Travel Skim Matrices Extracted for the Short Distance Personal Travel Model

Mode	Property	AM	PM	Mid	Off	Freeflow
SOV	Time	1	1	1	1	1
SOV	Distance	1	1	1	1	
SOV	Toll	1	1	1	1	
HOV2	Time	1	1	1	1	1
HOV2	Distance	1	1	1	1	
HOV2	Toll	1	1	1	1	
HOV3+	Time	1	1	1	1	1
HOV3+	Distance	1	1	1	1	1
HOV3+	Toll	1	1	1	1	
Walk	Time	From HOV3 freeflow distance				
Bike	Time	From HOV3 freeflow distance				
SchBus	Time	HOV3	HOV3	HOV3	HOV3	
TranWalk	In Time	1	1	1	1	
TranWalk	Out Time	1	1	1	1	
TranWalk	Fare	1	1	1	1	
TranAutoAccess	In Time	1	1	1	1	
TranAutoAccess	Out Time	1	1	1	1	
TranAutoAccess	Fare	1	1	1	1	
TranAutoEgress	In Time	1	1	1	1	
TranAutoEgress	Out Time	1	1	1	1	
TranAutoEgress	Fare	1	1	1	1	

12.5 Auto Skims

The 40 "auto skims" are extracted from the results of separate multi-class assignments run for each of the four model time periods (AM peak, PM peak, Midday and Off-peak).

Travel demand by time period for each of the vehicle modes used in the CSTDM (single occupant "auto" (SOV); high occupant "Auto" with 2 persons in the vehicle (HOV2); high occupant "auto" with 3+ persons in the vehicle (HOV3+); and truck) are assigned to the road network. A multi-class equilibrium assignment is run in CUBE for each time period, and resulting "loaded" travel times, distances and toll skims extracted. The assignment and skim process has the following features:

- Truck volumes are converted to passenger car equivalent (pce) volumes for assignment purposes, using an average truck pce factor of 1.5 for single-unit (medium) trucks and 2.0 for multi-unit (heavy trucks);
- The link USE field is used to identify availability of links for auto modes:
  - USE = 1 identifies links available for all autos
  - USE = 2 identifies links only available for autos with 2+ occupancy
  - USE = 3 identifies links only available for autos with 3+ occupancy:
  - USE = 5 and 6 - transit line / connector links NOT used in auto assignment
- The primary convergence criteria for the multi-class assignment is a "relative gap" < 0.01 (1%);
- A volume delay function BPR formula was used to calculate loaded link times for all road links except centroid connectors, of the form:
  - $TC = To * (1 + 0.18 (Volume / Capacity))^7$
- Intra-zonal distances are calculated as:
  - 1/3 * square root of zonal area (this formula was estimated empirically from data from census "On the Map" Journey to work data).

Intra-zonal auto times are calculated using the intra-zonal distance, using a speed value based on zonal density (defined as (Population + Employment) / Area in square miles). Intra-zonal speed classes are defined in Table 33. Intra-zonal auto times are capped at 10 minutes.

Table 33: Intra-Zonal Auto Speed Classification

Intra-Zonal Speed Class	Name	Minimum Density	Maximum Density	Auto Speed (mph)
1	Rural	0	200	50
2	Exurban	200	2,000	35
3	Suburban	2,000	20,000	25
4	Urban	20,000	n/a	15

Tolls for each origin-destination pair are obtained using the CUBE path skimming process for path-based tolls. Each toll is specified in the road network as a “closed system”, with an on-gate link, a toll-road link; and an off-gate link. By convention for the CSTDM, three consecutive links are identified at each location, and the same unique number coded in the relevant link ENTRYGATE, TOLL, OR EXITGATE field. In addition, an input database file is read in for each time period giving the tolls for autos and trucks for each toll location, with the following fields:

- ENTRYGATE (e.g. 1);
- EXITGATE (e.g. 1);
- Toll for Auto (e.g. $2.00);
- Toll for Truck (e.g. $10.00): the toll for 4-axle trucks is used to represent truck tolls;

The path-building process converts these tolls into “equivalent minutes” using values of time factors for each vehicle mode. Time-based Assignment paths for each O-D zone pair are then determined, taking into account the additional “equivalent time” penalty associated with the toll. The path skimming process then sums the $ value of all toll links on the shortest “time” path for each O-D pair. Value of time factors are “hard-coded” in the CUBE path-building script:

- 6 minutes per toll $ for SOV (single occupant vehicle auto);
- 3 minutes per toll $ for HOV2 (two occupant vehicle auto);
- 2 minutes per toll $ for HOV3 (3+ occupant vehicle auto, using 3.6 average occupancy);
- 3 minutes per toll $ for trucks.

Additional features used in the assignment and skimming process are:

- Walk and Bike skim times are derived from the HOV3+ distance skims, assuming an average walking speed of 2.5mph and an average bicycle speed of 11mph.
- Auto operating cost skims for the year 2000 are derived by multiplying the SOV auto distance skim by an average auto operating cost of 14 cents per mile (this operating cost per mile is taken from Table 2.1 of the August 2006 "Levels-of-Service Assumptions and Forecast Alternatives" Report prepared for the Bay Area / California High-Speed Rail Ridership and Revenue Forecasting Study, which gives historic information on typical Auto Operating Costs in California.
- Non-home destination zone parking costs for each O-D pair are added during the short distance personal travel model estimation process (see separate Technical Note on Parking Costs).

12.6 Transit Skims

The 36 "transit skims" identified in Table 32, used in the SDPTM are similar to the auto skims. In vehicle transit time, out of vehicle transit time, and transit fare are extracted for each of the three transit modes (transit with walk access AND egress; transit with auto access and walk egress; transit with walk access and auto egress), for each of the four model time periods (AM, PM, Midday, Offpeak).

A hybrid method is being used to represent local transit in the CSTDM. Rail services available for "local" travel are explicitly coded into the CUBE network. Local "bus service" is represented indirectly through a model relating bus times to auto time from CUBE. This model used for the representation of local bus transit is documented in a separate Technical Note. It is applied to a series of catchment area O-D pairs with a separate catchment area for each operator or group of operators serving regions or communities.

For local transit with walk access and egress 5 transit alternatives are available:

- Local bus all the way with walk access and egress (transit travel times and costs estimated using the indirect transit time model and the CUBE auto time);
- Local rail all the way with walk access and egress (transit travel times and costs extracted using CUBE);
- Local bus (with walk access) access to a rail station with walk egress (transit travel times and costs for Local bus component estimated using the indirect transit time model; rail and walk egress times and costs extracted using CUBE);
- Walk access to a rail station with local bus to destination (with walk egress) (transit travel times and costs for Local bus component estimated using the indirect transit time model; rail and walk access times and costs extracted using CUBE);
- Local bus (with walk access) access to a rail station with local bus to destination (with walk egress) (transit travel times and costs for Local bus component estimated using the indirect transit time model; rail times and costs extracted using CUBE).

Transit travel times and fares are extracted using CUBE for each of the five alternatives. The alternative with the lowest “generalized cost” is selected as the best alternative. Transit in vehicle time, out of vehicle time (including auto access / egress times as relevant), and travel cost (including transit fare and auto operating costs as relevant) are extracted for that alternative.

The generalised cost formulation for each alternative is in time units of equivalent in-vehicle time, and uses the following factors:

- Transit and auto in-vehicle time weighted by factor of 1.0;
- Transit walk, wait, transfer and access time weighted by 2.89 (this factor is taken from the factor estimated in the Sacramento SACSIM Activity-Based Travel Forecasting Model for SACOG (derived from Table 8 - Simplified Mode choice Model for calculating Aggregate Logsum, Technical Memo #4 Mode Choice Models, SACSIM05, August 2006)

- Transit fare costs converted to time units by applying a value of time of $8.21 per hour i.e. $1 is equivalent to 7.30 minutes (this value of time is also taken from the SACSIM model).

For local transit with auto access to a rail station and walk egress the following two transit alternatives are available:

- Auto to rail station and walk egress (transit travel times and costs extracted using CUBE);
- Auto to rail station and local bus to destination with walk egress (auto access and rail transit travel times and costs extracted using CUBE; transit travel times and costs for Local bus component estimated using the indirect transit time model).

For local transit with walk access to a rail station and auto egress to destination the following two transit alternatives are available:

- Walk access to rail station and auto egress (transit travel times and costs extracted using CUBE);
- Local bus access to rail station and auto egress (auto egress and rail transit travel times and costs extracted using CUBE; transit travel times and costs for Local bus component estimated using the indirect transit time model).

Skims for the best alternatives for the above two transit mode options are extracted as described above.

12.7 Travel Skims Extracted for Other CSTDM Model Components

The long distance personal travel model, the short distance commercial vehicle model, and the external vehicle model directly use the auto skims extracted.

The long distance personal travel model also uses transit skims for long distance rail and air service. The long distance rail skims are extracted directly from CUBE for two time periods (AM peak and Midday) using the explicit CUBE rail line coding. The long distance air skims are extracted directly from CUBE for one time period (AM peak) using the explicit CUBE air line coding.

Access and Egress skims to long distance rail and air by auto, local transit and walk are also required. These are also extracted from CUBE using the methods identified in Section 12.6 above.

The short distance commercial vehicle model, the long distance commercial vehicle model, and the external vehicle model also use truck skims, which are extracted in a similar fashion to the auto skims. Table 34 identifies the 14 truck skims extracted.

Table 34: Truck Travel Skim Matrices Extracted

Mode	Property	AM	PM	Mid	Off	Freeflow
Truck	Time	1	1	1	1	1
Truck	Distance	1	1	1	1	1
Truck	Toll	1	1	1	1	

13. Public Transport: Short Distance

13.1 Description

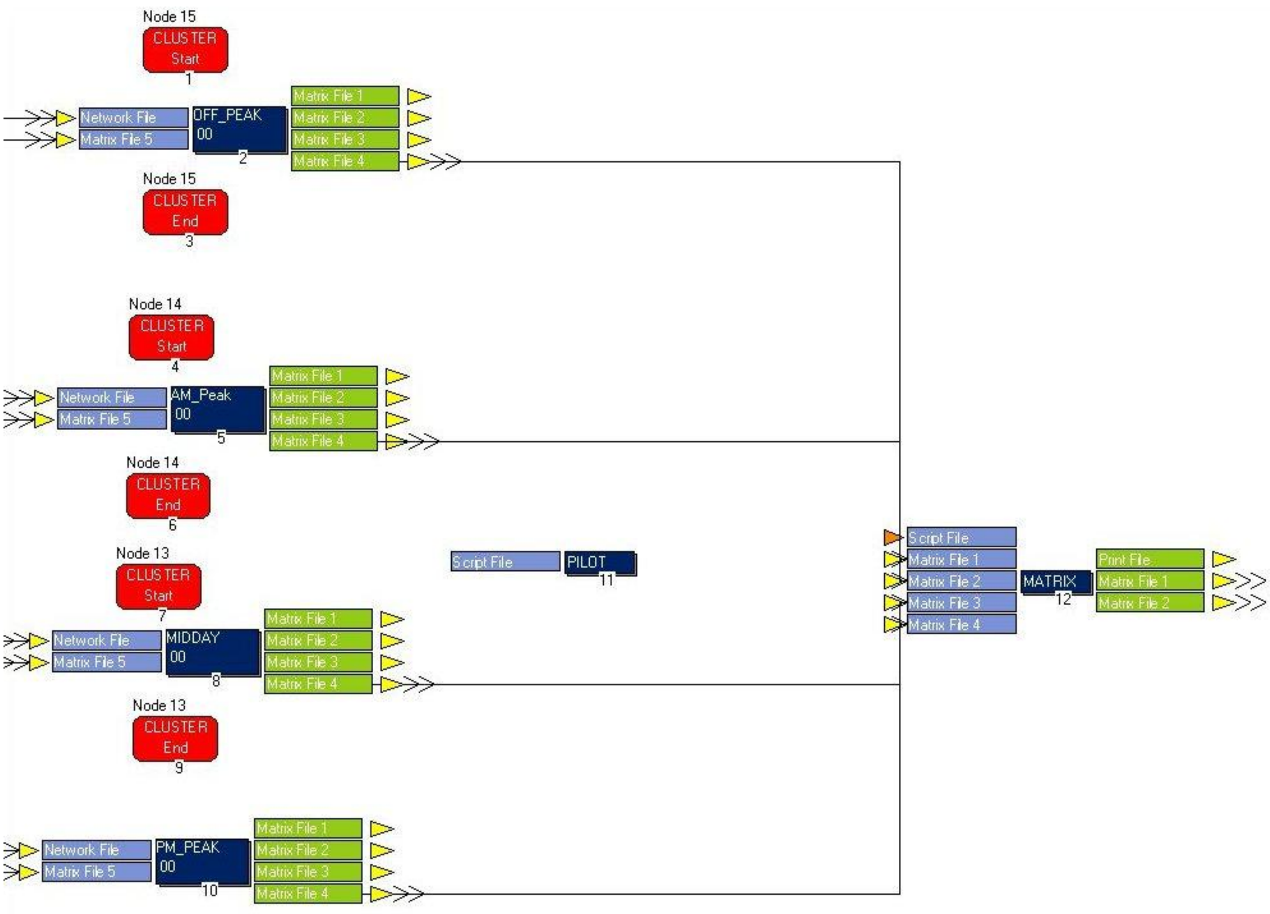


Figure 16: Short Distance Public Transport

Short distance public transport makes use of the loaded network and produces ridership on local fixed guideway rail, and bus services for each time period. The process flow is illustrated in figure 16.

13.2 Inputs

For each time period, the inputs include:

- Loaded network file <CSTDM_Directory>\base\<Scenario>\LoadedNetworks\HwyNetwork_Loaded_<time period>_<year>.NET
- Line files for each service: The process for creating these is described in the network documentation. The line files are found in <CSTDM_Directory>\base\<Scenario>\Controls\system_cstdm2009_<year>.pts Listings of the line files are in Tables 36 and 37.
- System files: they identify each transit system and the lines that apply to it. <CSTDM_Directory>\base\<Scenario>\transit\tfares\farein_cstdm2009_2008.far
- Fare file: This is a list of the fare system type for each transit system and cost if the system is a flat rate. If the system is a zone to zone system, then it contains the reference to the correct fare matrix. <CSTDM_Directory>\base\<Scenario>\transit\tfares\farein_cstdm2009_2008.far
- Fare Matrices: the zone to zone costs for each transit system that uses a zone to zone fare system. <CSTDM_Directory>\base\<Scenario>\transit\tfares\
- Factor File: This file sets properties for the transit assignment including the maximum number of transfers, wait curves, boarding and transfer penalties. <CSTDM_Directory>\base\<Scenario>\Controls\fact_cstdm2009_<year>.dat the factor file includes the factors described in table 35.

Table 35: Factors Included in the Factor File

Factor	Description	Value in CSTDM2009
FARESYSTEM	Number of fare model that applies to operators selected with OPERATOR keyword	11-16; 31-35; 41-43; 52; 54
SPREADFUNC	Integer of the function that computes SPREAD in route enumeration – SPREAD defines an upper cost limit for routes between an O-D pair	1 Max of Minimum O-D time * SPREADFACT; minimum O-D time + SPREADCONST
SPREADFACT	Factor that minimum O-D time is multiplied by	1.2
SPREADCONST	Time added to minimum O-D	0.0
MAXCOMPD	# components generated during route enumeration	1,000,000
MAXFERS	Maximum # transfers between O-D	2
AONMAXFERS	Maximum # transfers between O-D for O-D with only one enumerated route	2
EXTRAXFERS1	# of transfers at which program stops exploration less direct routes	2
EXTRAXFERS2	Max # of transfers explored in excess of the # required by minimum cost route	2
RUNFACTOR	Mode specific weighting factors applied to transit in-vehicle times and non-transit leg times in route enumeration	2.89 for modes 1-5 – walk and auto access and egress plus walk transfer
BRDPEN	Mode specific boarding penalty in minutes (for transit modes only)	5 for all transit modes
WAITFACTOR	Node specific wait-time weighting factor	2.89 for all non-TAZ nodes
IWAITCURVE	Wait curve used to calculate initial wait time for nodes specified by NODES keyword	1 for all non-transit nodes except 18501-18518 4 for nodes 18501-18518 (Air)
XWAITCURVE	Wait curve used to calculate transfer wait time at nodes	Same as IWAITCURVE
XFERPEN	Transit mode to transit mode transfer penalty, in minutes.	10, for all transit mode transfers

At present, line files are in use for the systems with fare structure reported in Table 36 and Table 37.

Table 36: Line Files for 2008 with Fare Structure

Transit Line File	Fare Structure Type
BART_CSTDM09_2008.lin	FROMTO
SACOG_LRT_CSTDM09_2008.lin	FLAT
SANDAG_LRT_CSTDM09_2008.lin	FLAT
VTA_LRT_CSTDM09_2008.lin	FLAT
Muni_Metro_CSTDM09_2008.lin	FLAT
SCAG_Urban_Rail_CSTDM09_2008.lin	FLAT
SANDAG_Sprinter_CSTDM09_2008.lin	FLAT
SANDAG_Rail_CSTDM09_2008.lin	COUNT
Metrolink_Orange-CSTDM09_2008.lin	FROMTO
SCAG_Metrolink_Other_CSTDM09_2008.lin	FROMTO
ACE_CSTDM09_2008.lin	FROMTO
CALTRAIN_CSTDM09_2008.lin	COUNT
Pacsurf_CSTDM09_2008.lin	FROMTO
AMTRAK_Capital_CSTDM09_2008.lin	FROMTO
AMTRAK_SJQ_CSTDM09_2008.lin	FROMTO

Table 37: Line Files for 2000 with Fare Structure

Transit Line File	Fare Structure Type
BART_CSTDM09_2000.lin	FROMTO
SACOG_LRT_CSTDM09_2000.lin	FLAT
SANDAG_LRT_CSTDM09_2000.lin	FLAT
VTA_LRT_CSTDM09_2000.lin	FLAT
Muni_Metro_CSTDM09_2000.lin	FLAT
SCAG_Urban_Rail_CSTDM09_2000.lin	FLAT
SANDAG_Rail_CSTDM09_2000.lin	COUNT
Metrolink_Orange-CSTDM09_2000.lin	FROMTO
SCAG_Metrolink_Other_CSTDM09_2000.lin	FLAT
ACE_CSTDM09_2000.lin	FROMTO
CALTRAIN_CSTDM09_2000.lin	COUNT
Pacsurf_CSTDM09_2000.lin	FROMTO
AMTRAK_Capital_CSTDM09_2000.lin	FROMTO
AMTRAK_SJQ_CSTDM09_2000.lin	FROMTO

13.3 Outputs

The outputs for each time period are four matrices each with separate subpages or skims. Each of these skims contains all of the zone to zone costs. In cases where no access is possible the value of -99999.00 is used to represent a null value. Each of these matrices can be found in

<CSTDM_Directory>\Base\<Scenario>\Skims\Transit\<time period>\.

In the matrix and skim names the following conventions are used:

W = Walk, R=Ride fixed route transit (light or heavy rail), T=Transit using the synthetic local transit, D=drive. These values are combined to create patterns such as "WRW" which indicates that the skim contains the times for Walking to a Ride trip and then Walking to the final destination.

- W_W_<time period>.mat: This matrix contains the walk access/walk egress skims of times for the following use patterns:

  - WRW: Walk to rail then walk to final destination
  - WRT: Walk to rail then transit to final destination
  - TRW: Transit to rail then walk to final destination
  - TRT: Transit to rail then transit to final destination
  - TTT: Transit for the entire trip
- D_W_<time period>.mat: This matrix contains the walk access/walk egress skims of times for the following use patterns:
  - DRW: Drive to rail then walk to final destination
  - DRT: Drive to rail then transit to final destination
- W_D_<time period>.mat: This matrix contains the walk access/walk egress skims of times for the following use patterns:
  - WRD: Walk to rail then drive to final destination
  - TRD: Transit to rail then drive to final destination
- Costs_<time period>.mat: This matrix contains the walk access/walk egress skims of times for the following use patterns:
  - WW_TIVT: Walk access/walk egress time in vehicle
  - WW_TOVT: Walk access /walk egress time out of vehicle
  - WW_TFARE: Walk access/walk egress total fare
  - DW_TIVT: Drive access/walk egress time in vehicle
  - DW_TOVT: Drive access/walk egress time out of vehicle
  - DW_TFARE: Drive access/walk egress total fare
  - WD_TIVT: Walk access/drive egress time in vehicle
  - WD_TOVT: Walk access/drive egress time out of vehicle
  - WD_TFARE: Walk access/drive egress total fare

13.4 Public Transport Line Management

Public transport scenarios and network updates can be created in a rather straightforward way in CSTDM. Appendix B provides an example that guides through the modifications of line files for public transportation. Updates introduced in the line files need to be accompanied by the necessary updates also in the related CSTDM input files used for public transportation described in the Technical Note on “Networks”.

As described in the Technical Note on “Local Transit Functions”, one of the advantages of the use of the synthetic methodology for the representation of local bus transit is the easier maintenance required for the update of the local transit network. Appendix C provides a simple example of the process required to update the information for the local transit component used in the SDPTM.

14. Public Transport: Long Distance

14.1 Description

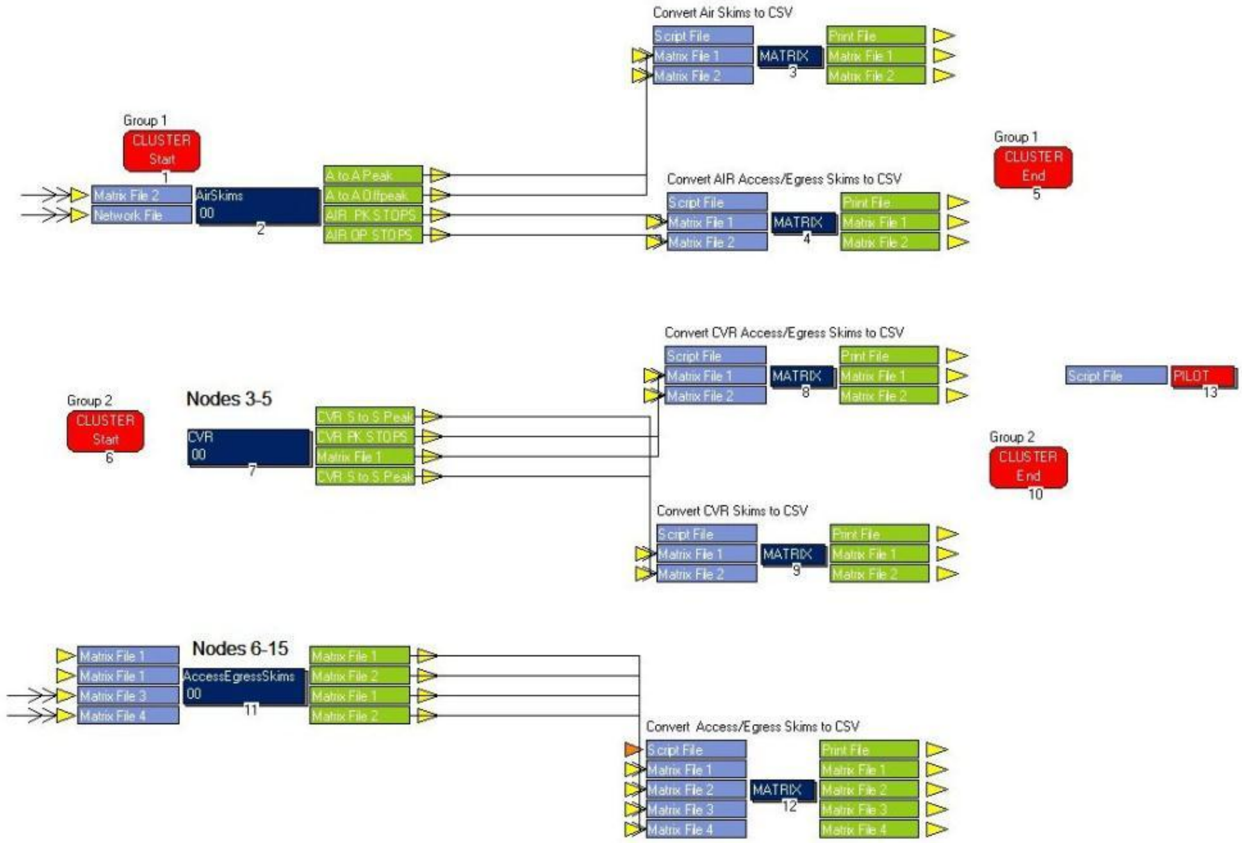


Figure 17: Long Distance Public Transport

The long distance public transport component of the CSTDM addresses the use of air travel and conventional rail for passenger inter-regional trips. Air skims and conventional rail skims are prepared separately and are run in parallel with each other to save time as shown in figure 17.

14.2 Inputs

The air skims component makes use of the line files for the ten intra-regional rail services, the fare matrices that apply, a fare file that identifies the fare structure for each of the access modes, and reuses the same system file as it is used for the short distance public transport calculations. The line files used for the air and rail access/egress skimming are listed in table 38.

Table 38: Line Files used for Air and Rail Skims

Service	**Line file**
BART	Bart_CSTDM09_<year>.lin
SacRT	SACOG_LRT_CSTDM09_<year>.lin
San Diego Metropolitan Transit Authority (Trolley)	SANDAG_LRT_CSTDM09_<year>.lin
Santa Clara Valley Transit Authority	VTA_LRT_CSTDM09_<year>.lin
San Francisco Municipal Transit Agency	Muni_metro_CSTDM09_<year>.lin
LA Metro	SCAG_Urban_Rail_CSTDM09_<year>.lin
San Diego Coaster	SANDAG_Rail_CSTDM09_<year>.lin
Metrolink	SCAG_Metrolink_Other_CSTDM09_<year>.lin
Caltrain	Caltrain_CSTDM09_<year>.lin
Air service	Air_CSTDM09_<year>.lin

The conventional rail component uses all of the line files from the short distance public transport. The system file, fare file, and fare matrices are applied unchanged from the short distance personal travel model.

The access/egress skims process for air and conventional rail uses auto skims for AM peak and midday time with time, tolls, distance and cost for transit.

- Auto Skims: <CSTDM_Directory>\Base\<Scenario>\Skims\Auto\Skims_<timeperiod>_<year>.mat
- Transit cost: CSTDM_Directory>\Base\<Scenario>\Skims\Transit\<time period>\Costs_<time period>_<year>.mat

14.3 Outputs

Outputs from the long distance public transport include matrices as .csv files for peak and off peak times for:

- Airport to airport in vehicle times, fares, headways and reliabilities: <CSTDM_Directory>\Models\LDPTM\<Scenario>\Inputs\Airport_to_airport_<time period>_<year>.csv
- Air access and egress skims: <CSTDM_Directory>\Models\LDPTM\<Scenario>\Inputs\Air _<time period>_stops_<year>.csv
- Conventional rail access and egress skims: CSTDM_Directory>\Models\LDPTM\<Scenario>\Inputs\CVR _<time period>_stops_<year>.csv
- Conventional rail times, fare, headway and reliability: <CSTDM_Directory>\Models\LDPTM\<Scenario>\Inputs\Station_to_station_<tim e period>_<year>.csv
- Access/egress skims: <CSTDM_Directory>\Models\LDPTM\<Scenario>\Inputs\ <time period>_Access_egress_<year>.csv
- Access/egress skims for conventional rail: <CSTDM_Directory>\Models\LDPTM\<Scenario>\Inputs\ <time period>_Access_egress_CVR_<year>.csv

14.4 Public Transport Line Management

Public transport scenarios and network updates can be created in a rather straightforward way in CSTDM. Appendix B provides an example that guides through the modifications of line files for public transportation. Updates introduced in the line files need to be accompanied by the necessary updates also in the related CSTDM input files used for public transportation described in the Technical Note on “Networks”.

15. Export to HDF5

15.1 Description

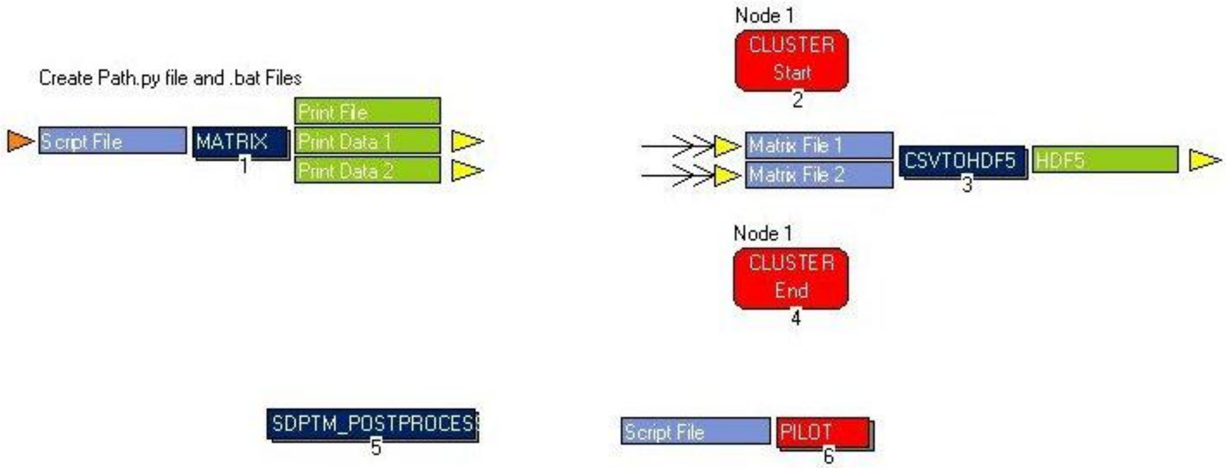


Figure 18: HDF5 Export and SDPTM Post Processor

The export to HDF5 process compiles all skims from the model and exports them to a single HDF5 file for use in the next run iteration. This is a python application with two steps as shown in figure 18. The first creates
<CSTDM_Directory>\models\HDF5\CSV\prepskims.bat and
<CSTDM_Directory>\models\HDF5\CSV\path.py.
The second executes the .bat file. An additional step runs in parallel and produces some post processed results for the short distance personal travel model.

15.2 Inputs

All skims for the CSTDM run.

15.3 Outputs

<CSTDM_Directory>\base\<Scenarios>\skims\skims.h5.

Appendix A

The following section describes simple instructions useful for some process of updates of the network links. This section is specifically designed to provide some explanatory cases, and does not include all possible changes that could be introduced in the model networks. Due to the complexity of the CSTDM model, a broad variety of possible cases for network updates and modifications is possible, and the user/developer can develop specific cases using predefined CUBE commands, specific scripts and additional integration with external software packages. No user manual can cover all possible applications, but this and the following Appendices provide a very brief introduction and demonstration of some of these capabilities.

The following figures present the CUBE Windows and commands used for some basic network editing procedures. Figure 19 contains a screenshot with the CUBE Network window in which the road network layers are visible.

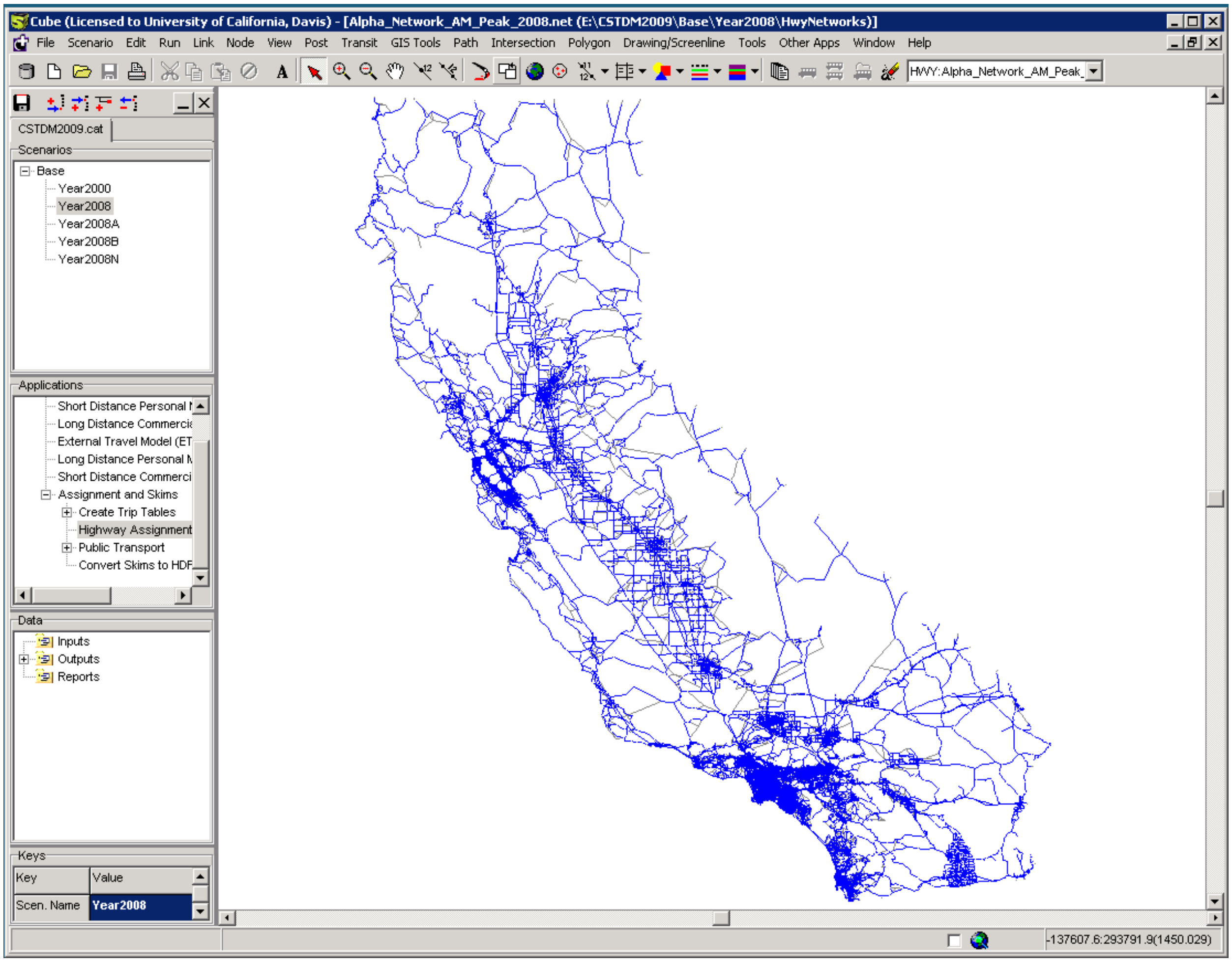


Figure 19: CUBE Network Window

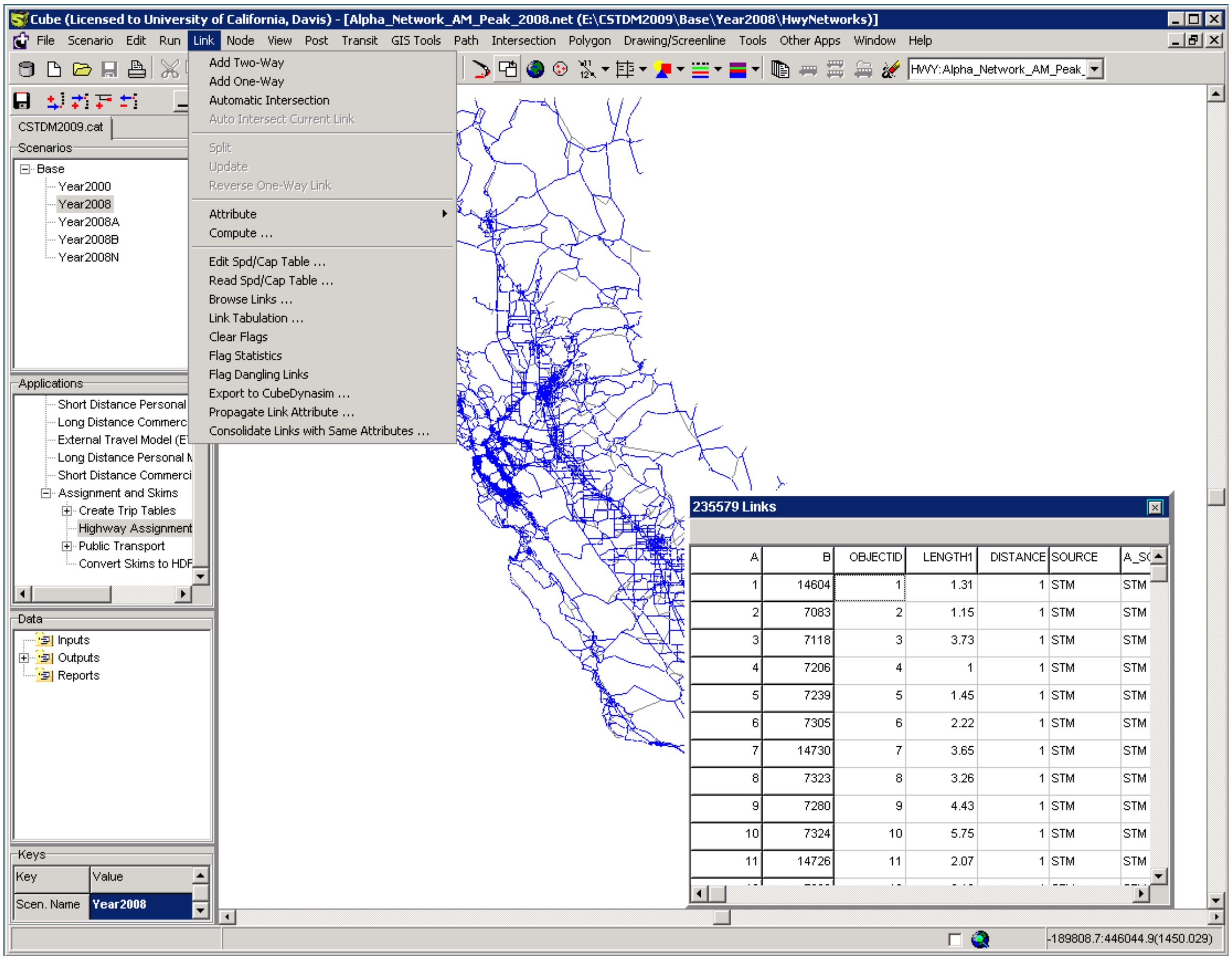


Figure 20: CUBE "Browse Link" Command

The command:

Link > Browse Links (Browse nodes)

opens a dialog box listing nodes in the road network. Figure 20 shows a screenshot with the "Browse Link" mode.

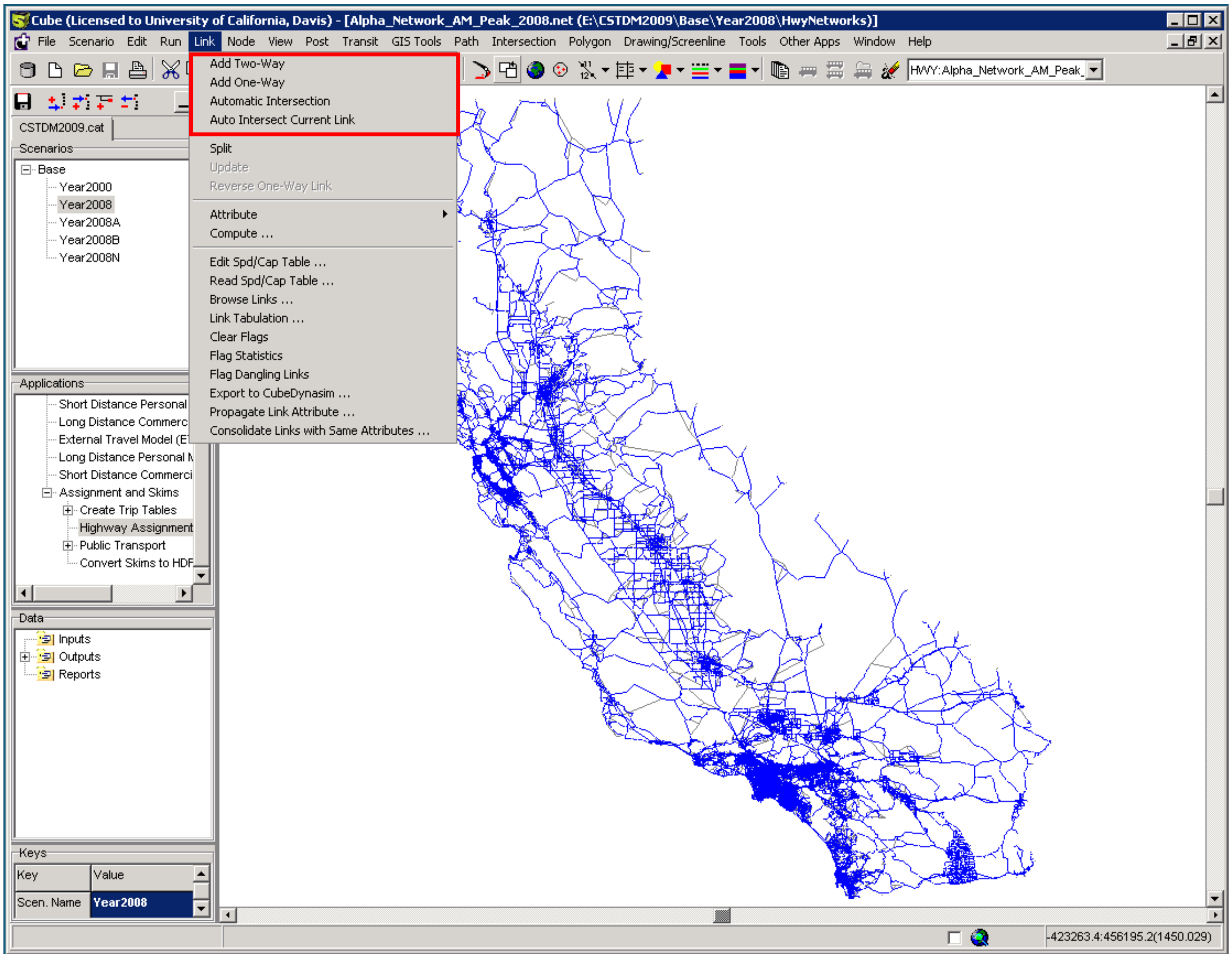


Figure 21: "Add a Link" Command

Figure 21 shows a single sequence to add a link to the network. Select the command:

Link > Add Two-Way Link

Or

Link > Add One-Way Link

To add a link to the network or introduce a new road to the network.

Figure 22 shows the process to Add/Edit centroid connectors in the road network.

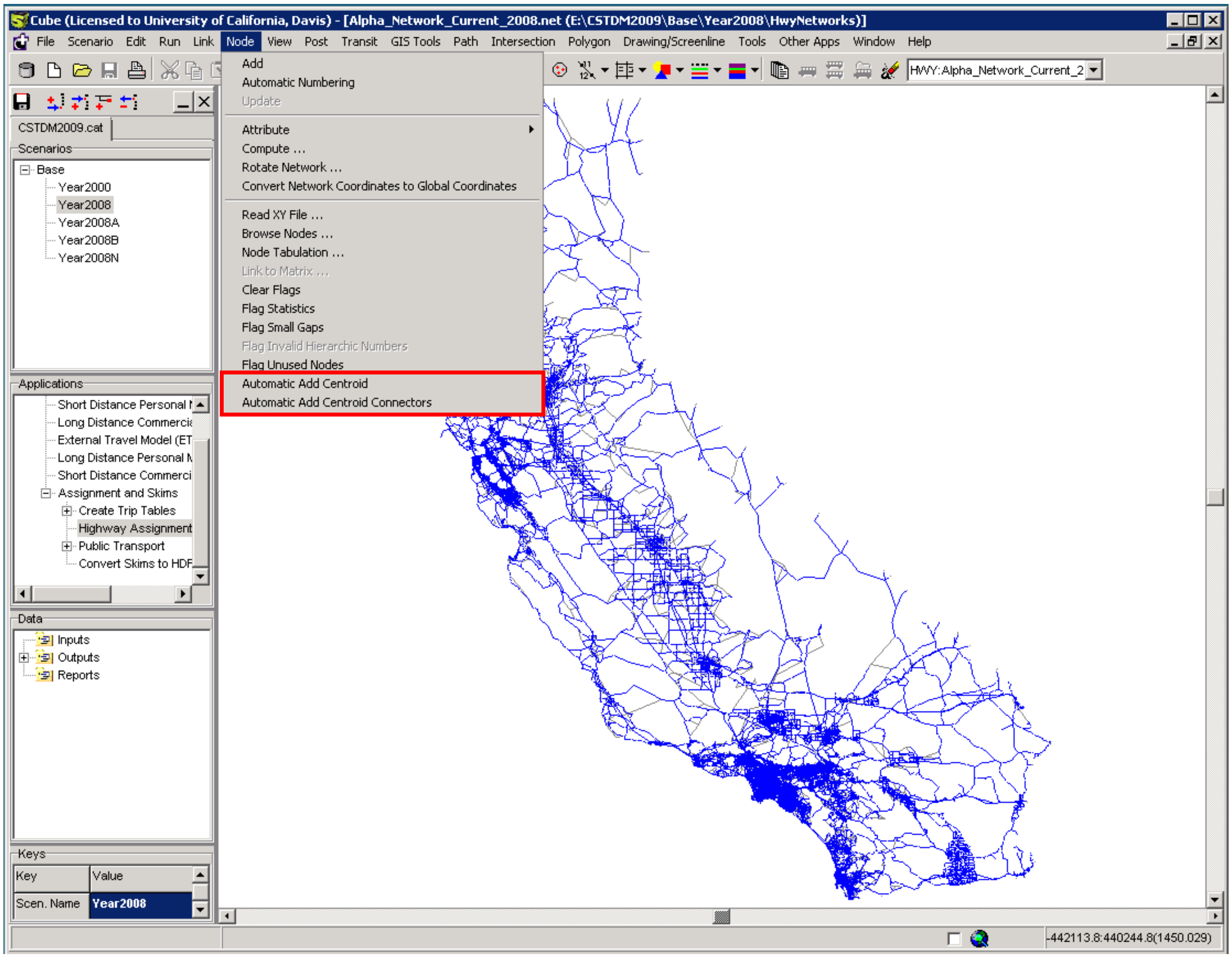


Figure 22: "Add/Edit Centroid" Command

After the process of editing is completed, the command "Save Log File" can be used to save the log file correspondent to the introduced changes (see Figure 23. Besides, please refer to the Technical Note on Networks for the use of Log File in CUBE to create/update Networks).

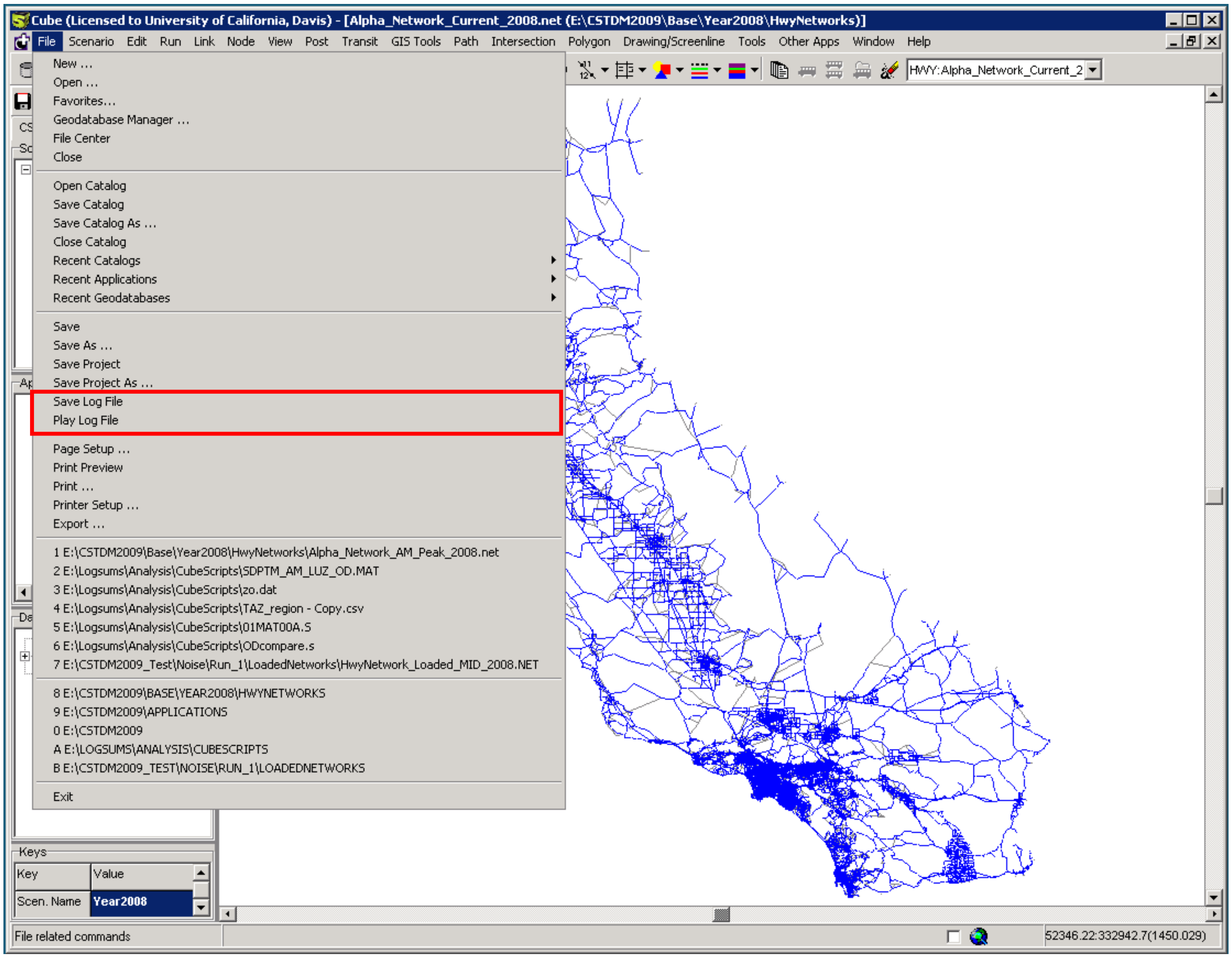


Figure 23: "Save Log File/Play Log File" Command

Appendix B

The following figures present a simple example involving the update of the Public Transportation Line Files.

Section 1: Adding a new transit station to the network

To begin with, the user needs to add a node that represents the transit station. Figure 24 displays the 3 steps required to add a node to the network. If the transit operator that uses this station uses a FROMTO or COUNT fare structure, a node attribute will need to be added. In the case of a FROMTO fare structure, a unique node attribute will be used to identify each station. For operators who use a COUNT fare structure, the node attribute will be used to identify what fare zone the station is in.

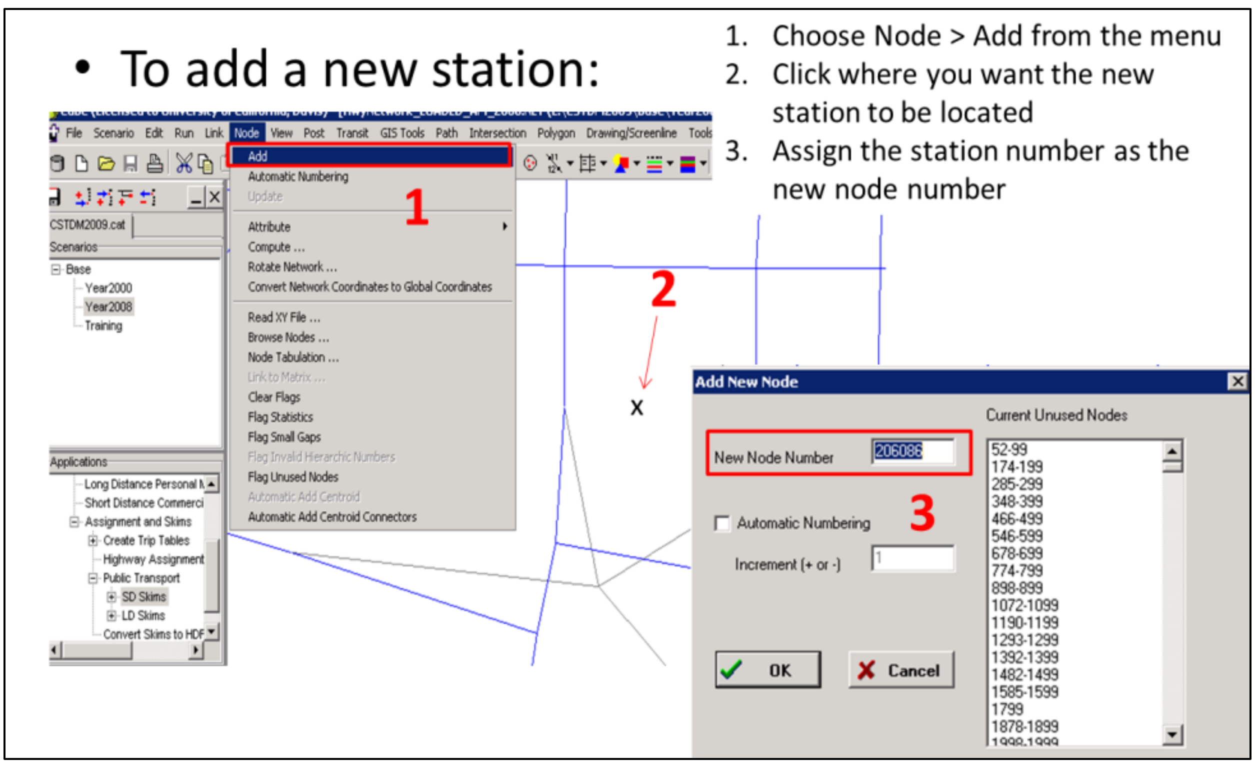


Figure 24: Adding a new station to the road network in CUBE

Once a node has been added to the network, a special link (special link refers to the unique link attributes) needs to be added to connect the station to the road network. Figure 25 displays the 3 steps required to do this.

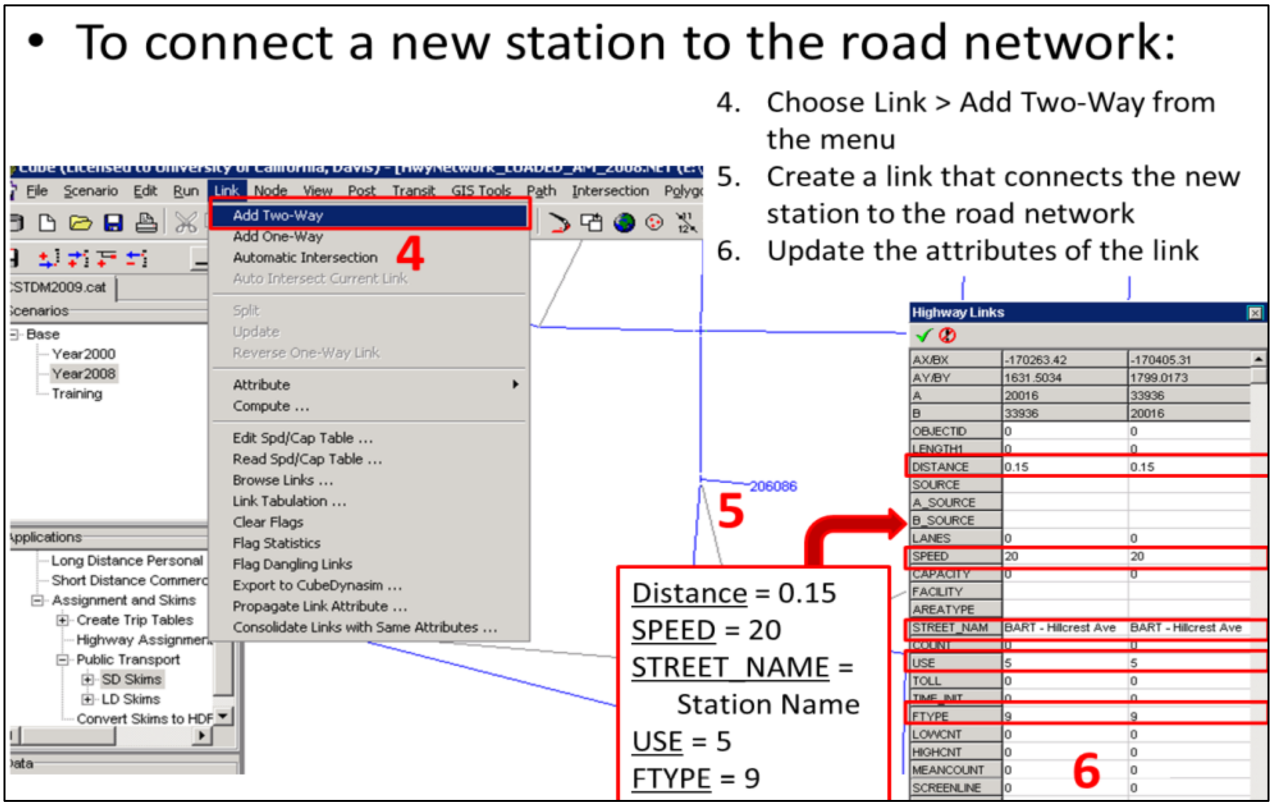


Figure 25: Connecting a node representing a transit station to the road network

Section 2: Creating rail links

Now that a station has been created and connected to the network, the stations along the transit line need to be connected. This is accomplished by adding a special link (special link refers to the unique link attributes) that represents the straight line connection between stations. Figure 26 displays the 3 steps required to create these links.

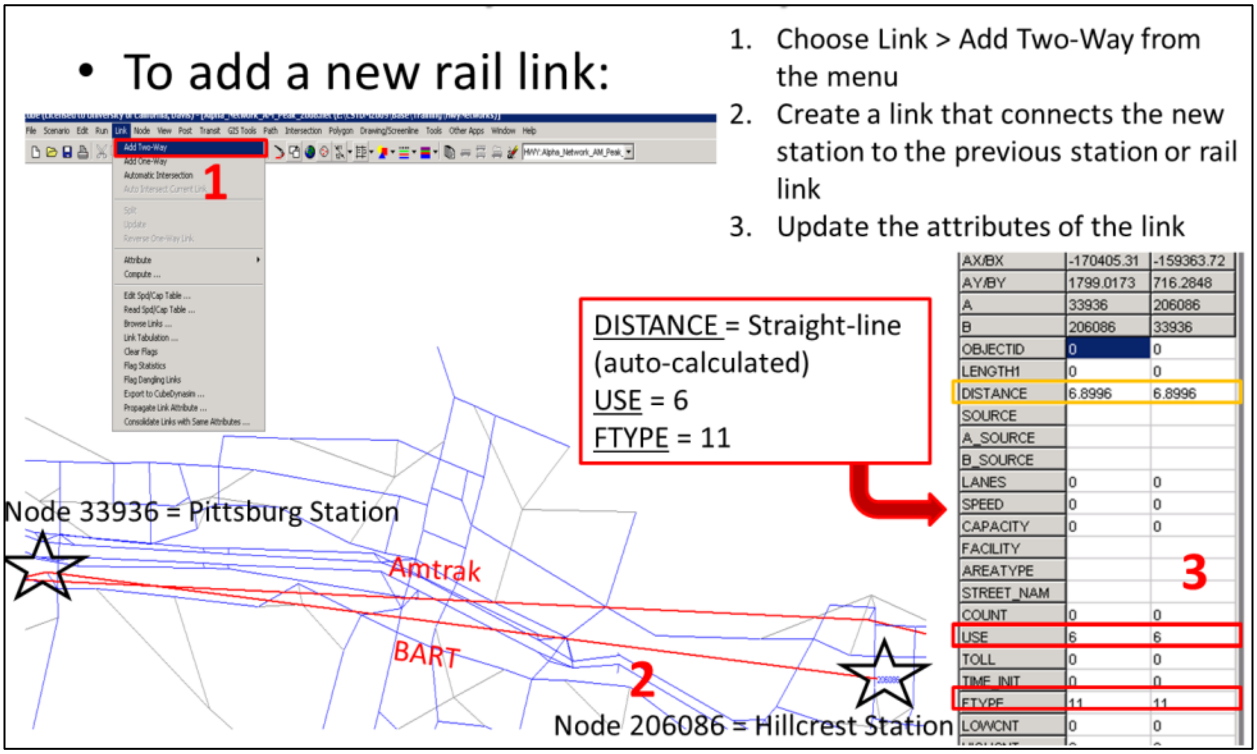


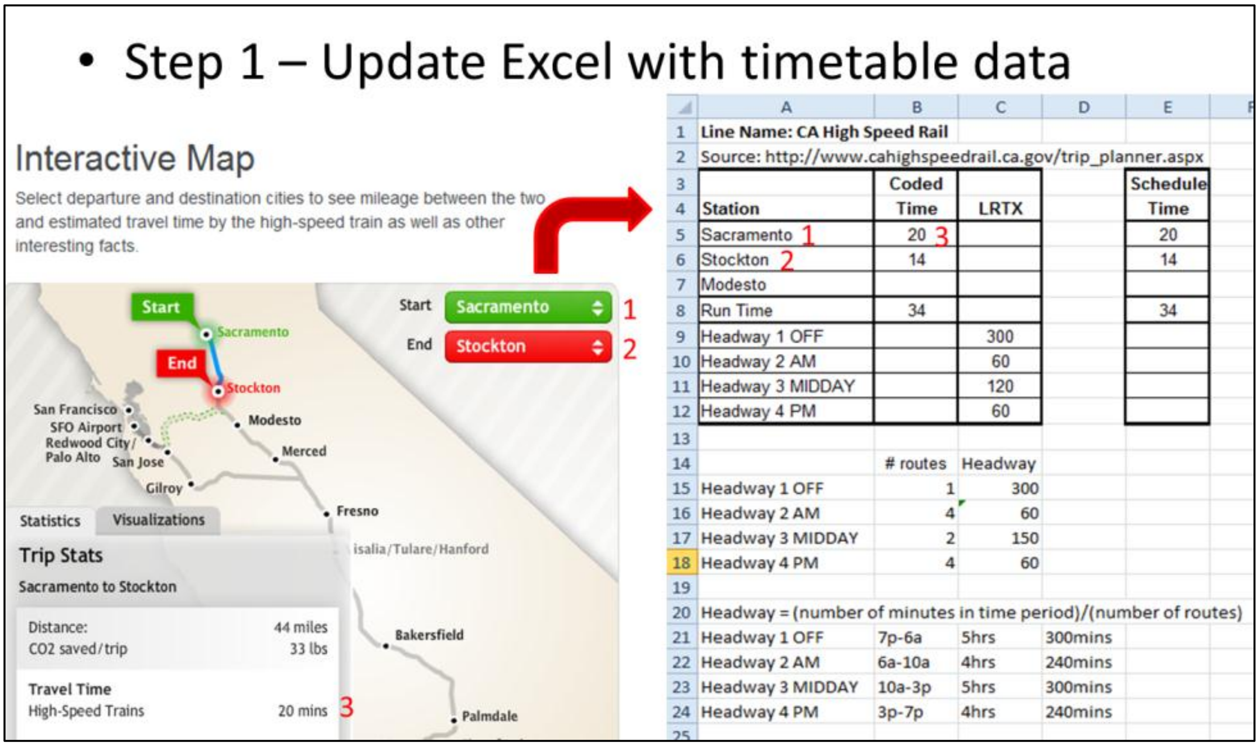
Figure 26: Connecting stations along a transit route

Section 3: Creating Line Files

In order for CUBE to use the newly created stations and rail links, a transit line file needs to be created. In the case that the user is updating an existing transit route (for example, a new station has been added to an existing route), the existing transit line file needs to be updated. In this section, the process of adding a new transit line file will be explained for a simple route with only 3 stations, but the process to modify an existing transit route uses the same steps (just fewer of them).

<u>Step 1: Update MS Excel with timetable data.</u>

Timetable data can be acquired from transit operators. From these timetables, the user will want to determine 2 things: the headway of the transit route in the 4 time periods, and the time between stations along the route. Once these have been determined, it is recommended to organize this information in MS Excel in a format that mimics the format that CUBE requires. Figure 27 displays an example of this process.



	A	B	C	D	E
1	Line Name: CA High Speed Rail				
2	Source: http://www.cahighspeedrail.ca.gov/trip_planner.aspx				
3		Coded			Schedule
4	Station	Time	LRTX		Time
5	Sacramento 1	20 3			20
6	Stockton 2	14			14
7	Modesto				
8	Run Time	34			34
9	Headway 1 OFF		300		
10	Headway 2 AM		60		
11	Headway 3 MIDDAY		120		
12	Headway 4 PM		60		
13					
14		# routes	Headway		
15	Headway 1 OFF	1	300		
16	Headway 2 AM	4	60		
17	Headway 3 MIDDAY	2	150		
18	Headway 4 PM	4	60		
19					
20	Headway = (number of minutes in time period)/(number of routes)				
21	Headway 1 OFF	7p-6a	5hrs	300mins	
22	Headway 2 AM	6a-10a	4hrs	240mins	
23	Headway 3 MIDDAY	10a-3p	5hrs	300mins	
24	Headway 4 PM	3p-7p	4hrs	240mins	

Figure 27: Translating timetable data into CUBE transit line file format

<u>Step 2: Update MS Excel with Fare Data</u>

Fare information can be acquired from the transit operators. There are 3 main types of fare structure: FROMTO, COUNT, and FLAT. For a detailed explanation of these 3 types, please see section 3.4 in the documentation on Networks (CSTDM09_Networks_Final.pdf). For transit authorities that use a COUNT (zone) system or a FLAT fare system, these fares need to be recorded so they can be added to the Fare File. Figure 28 displays how the 3 different types of fate structures are used within the CUBE Fare File.

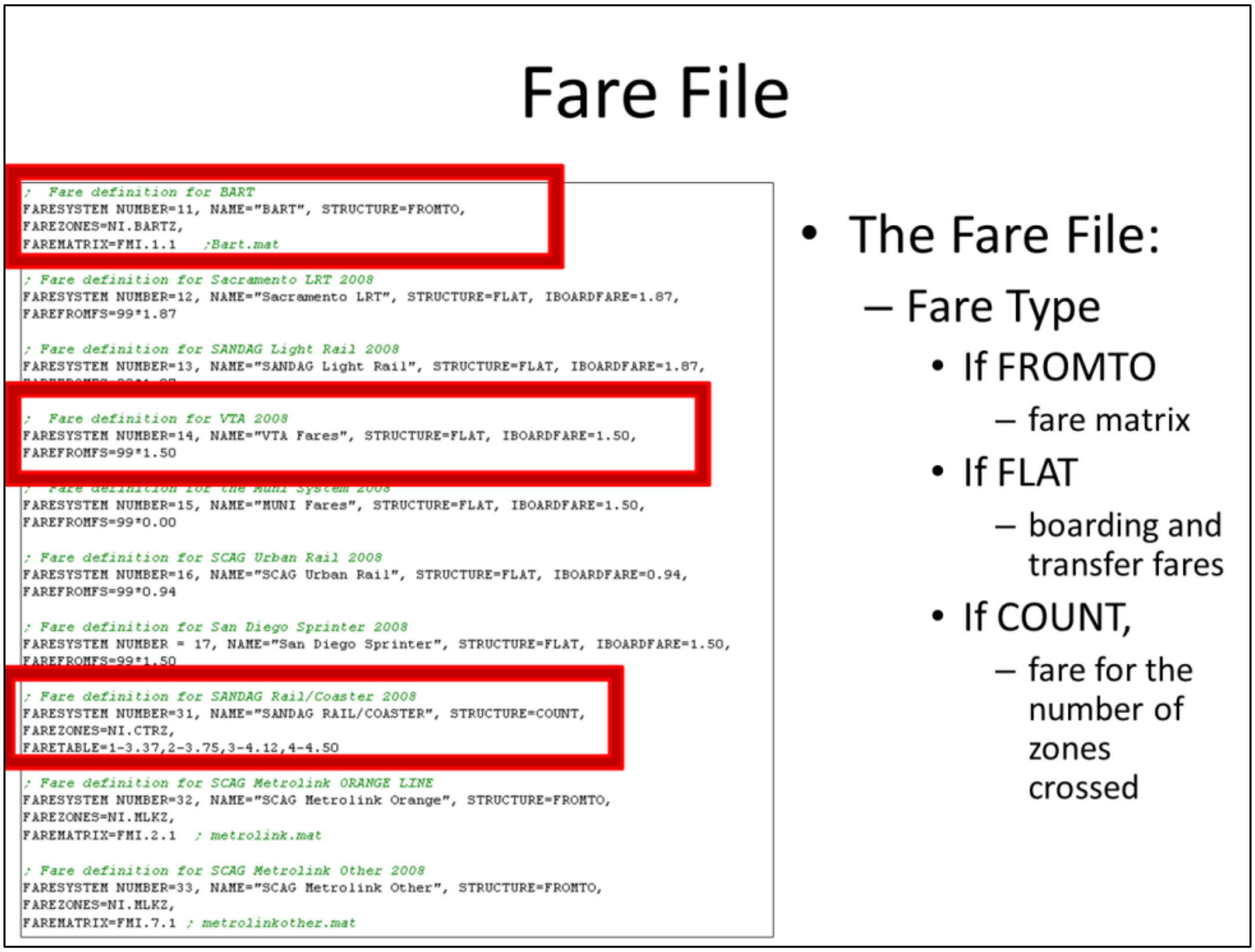


Figure 28: Examples of the 3 fare types in the CUBE Fare File

As you can see in Figure 28, COUNT and FLAT fare rates are simply added to the code. For operators that use a FROMTO fare structure, a matrix is referred to. Figure 29 provides an example of how to convert fares between stations to a fare matrix in MS Excel.

It is important to remember that CSTDM09 has all fares in 2000 US dollar equivalents. So if the user is adding fare data for 2008, it needs to be converted to 2000 dollars.

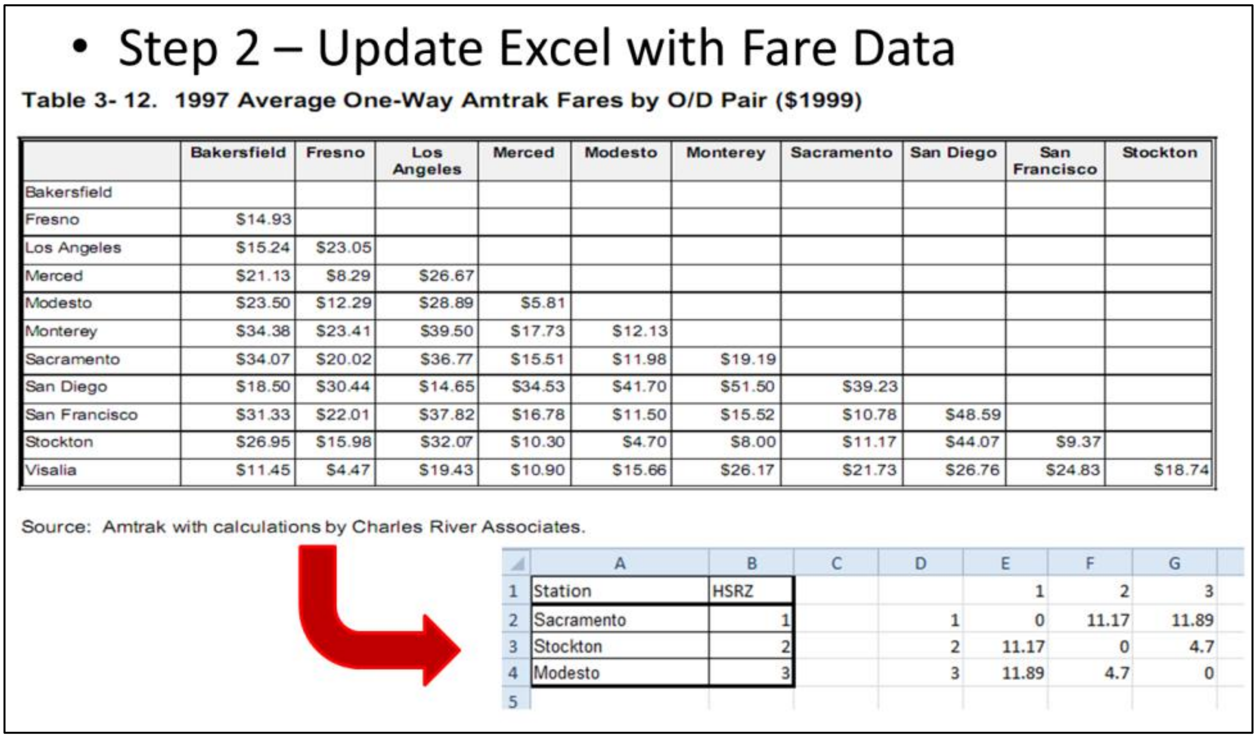


Table 3- 12. 1997 Average One-Way Amtrak Fares by O/D Pair ($1999)

	Bakersfield	Fresno	Los Angeles	Merced	Modesto	Monterey	Sacramento	San Diego	San Francisco	Stockton
Bakersfield										
Fresno	$14.93									
Los Angeles	$15.24	$23.05								
Merced	$21.13	$8.29	$26.67							
Modesto	$23.50	$12.29	$28.89	$5.81						
Monterey	$34.38	$23.41	$39.50	$17.73	$12.13					
Sacramento	$34.07	$20.02	$36.77	$15.51	$11.98	$19.19				
San Diego	$18.50	$30.44	$14.65	$34.53	$41.70	$51.50	$39.23			
San Francisco	$31.33	$22.01	$37.82	$16.78	$11.50	$15.52	$10.78	$48.59		
Stockton	$26.95	$15.98	$32.07	$10.30	$4.70	$8.00	$11.17	$44.07	$9.37	
Visalia	$11.45	$4.47	$19.43	$10.90	$15.66	$26.17	$21.73	$26.76	$24.83	$18.74

Source: Amtrak with calculations by Charles River Associates.

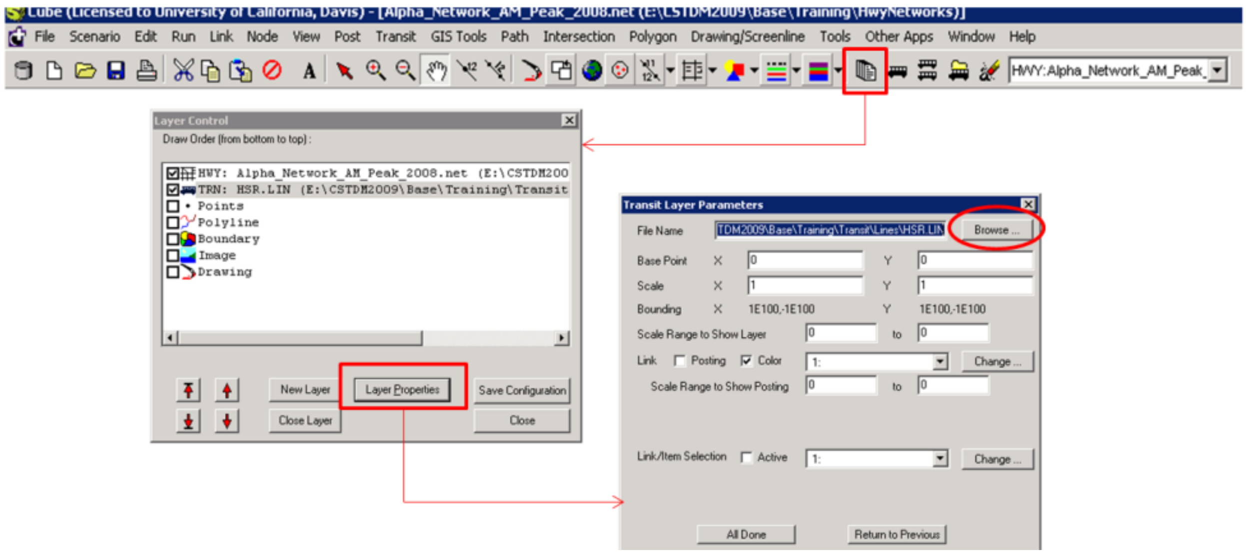
	A	B	C	D	E	F	G
1	Station	HSRZ			1	2	3
2	Sacramento	1		1	0	11.17	11.89
3	Stockton	2		2	11.17	0	4.7
4	Modesto	3		3	11.89	4.7	0
5							

Figure 29: Translating station to station fares to a fare matrix in MS Excel

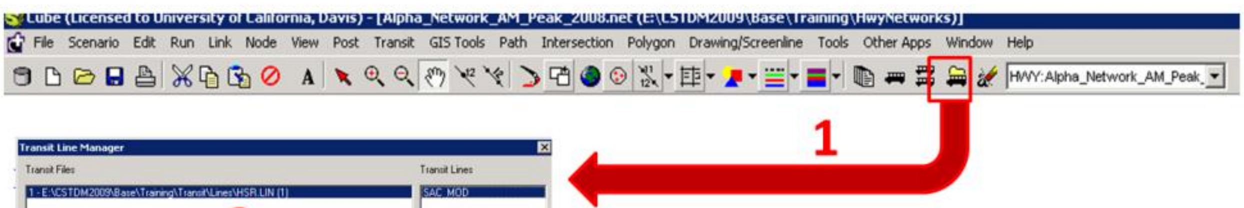
<u>Step 3: Create a new operator in CUBE</u>

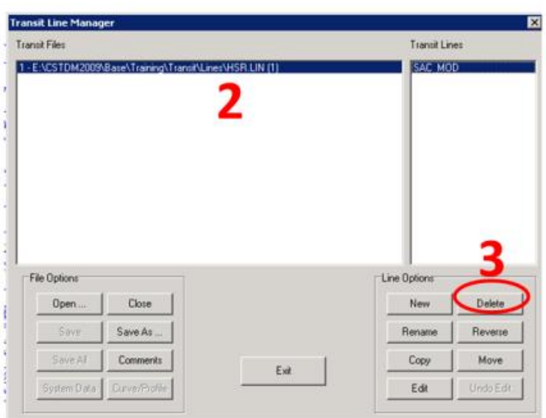
Once MS Excel has been used to translate timetable and fare information from the transit operator into formats that CUBE requires, it is time to start adding this information to CUBE. To begin, the user needs to create a new line file. There should only be one line file per transit operator. If there is more than one route that the transit operator offers, individual lines (routes) are added within the line file. The easiest way to add a new line file is to copy an existing line file to a new name, empty it, and then populate it with the information set up in steps 1 &2. Figure 30 explains how to do this.

- Step 3 – Create a new operator in CUBE
  a. Copy an existing line file and rename to ***.lin in Windows Explorer
     a. *** = a unique name for the new operator
  b. Open CUBE, add the ***.lin file as the Transit line within Layer Control (see below)



c. Delete the existing lines within the HSR line file





1. Click on the Transit Line Manager button ( )
2. Highlight an existing line
3. Click on the Delete button
4. Repeat steps 2 & 3 until all lines are deleted

Figure 30: Creating a new operator (line file) in CUBE

Step 4: Create a new line (route) in CUBE

For each unique route (line) that an operator provides, a line needs to be added to the line file. In our simple example with 3 stations we only have one line, but this process can be repeated for as many lines an operator provides. Figure 31 displays how to create a new line and then edit the contents.

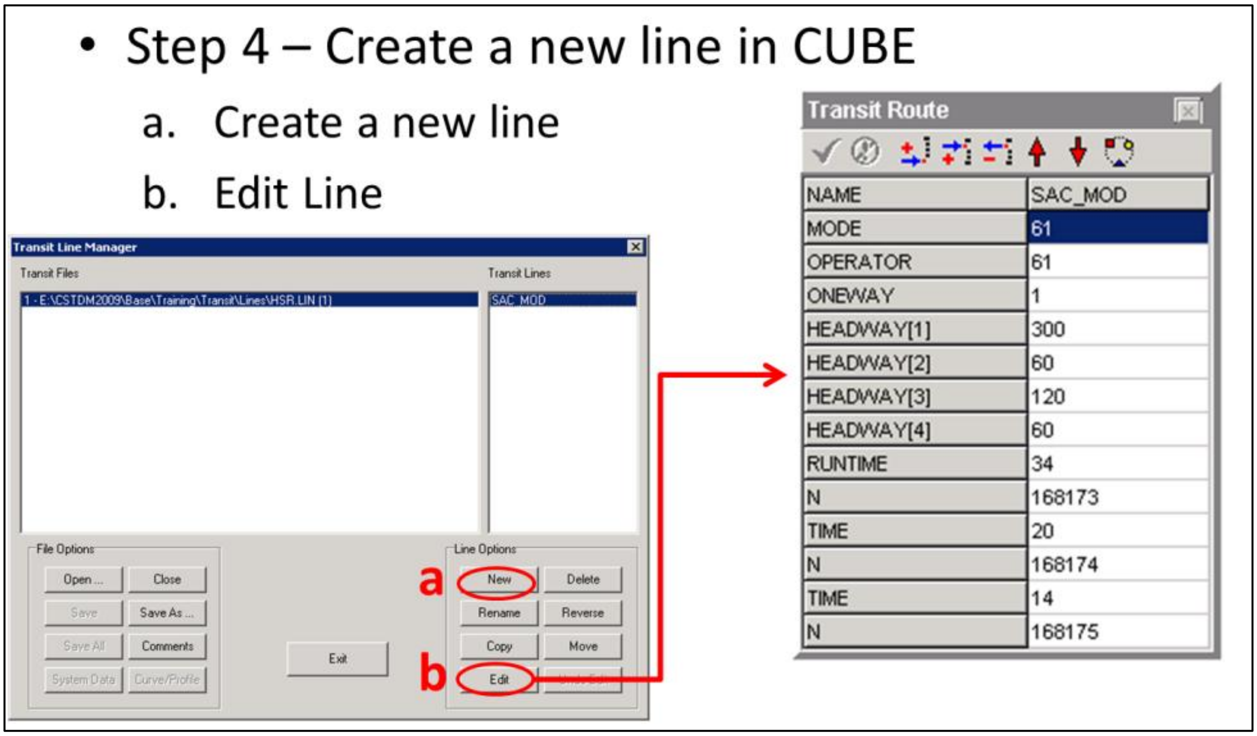


Figure 31: Create a new line within a line file

Now that a line file has been created for the operator with at least one line (route), 3 scripts within CUBE need to be updated to allow the new operator to be available for transit trips. These three files are the System File, Factor File, and Fare File.

Step 5: Edit the System File

The System File requires two additions for a new operator and one optional addition. These additions can be seen in Figure 32 below. The first is to add a new OPERATOR and give it a NUMBER (unique ID) and a NAME (1 in Figure 32). Next, the same NUMBER and NAME needs to be added as a MODE (2 in Figure 32). For simplicity, we have coded all MODEs and OPERATORs with the same NUMBER and NAME. Number

3 in Figure 32 is optional. If the new operator uses a different wait curve then the existing transit operators, a new wait curve can be added to the System File. Wait curves are linked to operators in the Factor File (explained in Step 6 below).

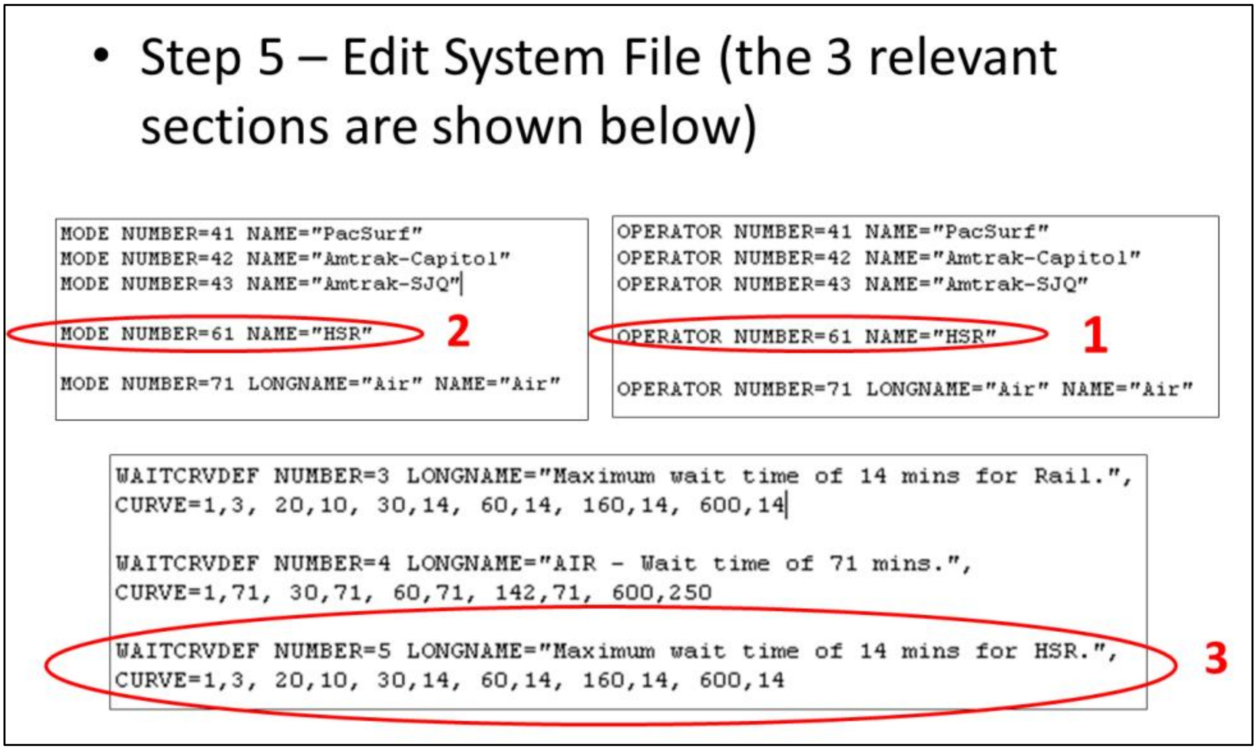


Figure 32: Editing the System File for new operators

<u>Step 6: Edit the Factor File</u>

The Factor File requires three additions for a new operator. These additions can be seen in Figure 33 below. The first is to link the new OPERATOR to a FARESYSTEM (1 in Figure 33). A new FARESYSTEM number should be added here and noted so that the user can input it during the next step (editing the Fare File). Next, a VALUEOFTIME needs to be assigned to the OPERATOR (2 in Figure 33). Finally, the node numbers (stations) that are used in the line file need to be added to a wait curve, both IWAITCURVE and XWAITCURVE (3 in Figure 33). As you can see in Figure 33, this can be ranges of node numbers or individual nodes separated by commas.

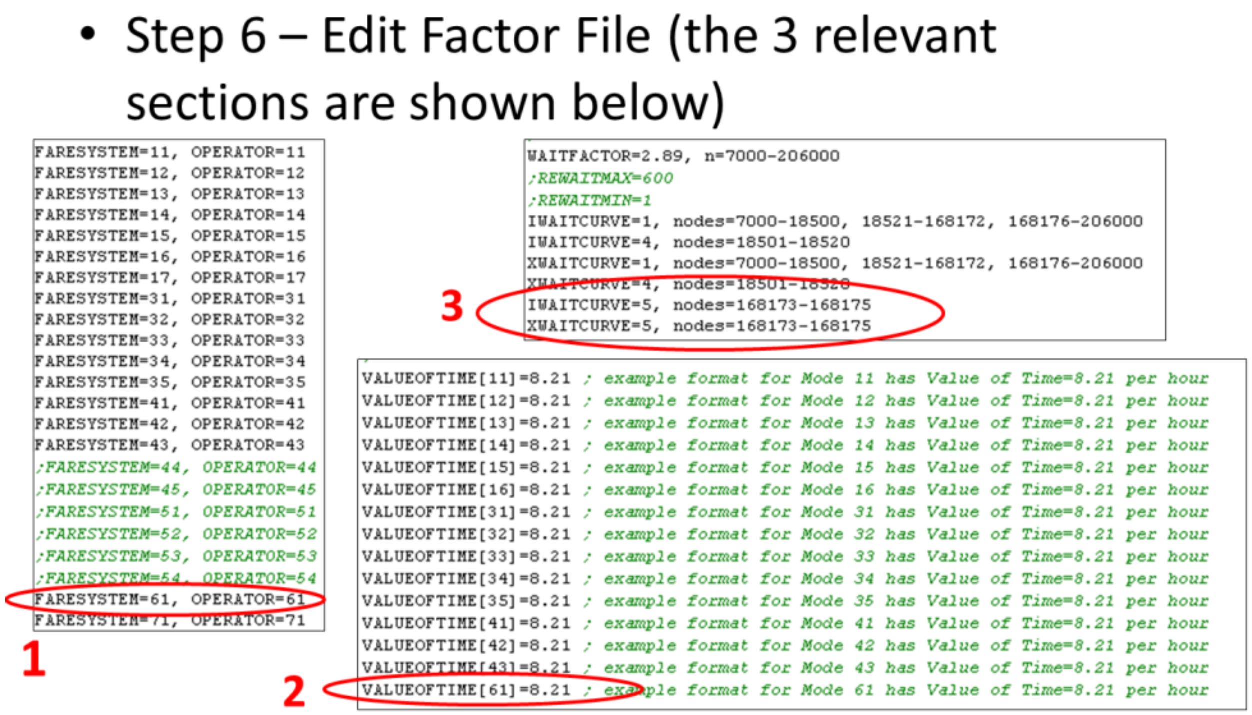


Figure 33: Editing the Factor File for new operators

<u>Step 7: Edit the Fare File</u>

The Fare File requires 1 addition for a new operator. These additions can be seen in Figure 34 below for operators that use a FROMTO fare structure and in Figure 28 for ones that use FLAT or COUNT fare structures.

FROMTO (Figure 34): In the first line of code the FARESYSTEM NUMBER, operator NAME, and STRUCTURE=FROMTO need to be added. On the second line of code a FAREZONES variable needs to be set that links the fare matrix to the nodes (stations) in the network. In the third line of code, the FAREMATRIX needs to be linked to the new operator. Creating a FAREMATRIX and adding it to CUBE are explained in the next two steps (steps 8 & 9).

FLAT (Figure 28): In the first line of code the FARESYSTEM NUMBER, operator NAME, STRUCTURE=FLAT, and the IBOARDFARE need to be added. IBOARDFARE is the fare the operator charges to board the transit system in dollars. On the second line of code the transfer fare amount needs to be added in the following format: FAREFROMFS=99*<Transfer Fare in dollars>.

COUNT (Figure 28): In the first line of code the FARESYSTEM NUMBER, operator NAME, and STRUCTURE=COUNT need to be added. On the second line of code a FAREZONES variable needs to be set that links the fares in the third line of code to the access\egress nodes (stations) used on the trip. In the third line of code the fare charged for the number of zones crossed needs to be entered in the following format: FARETABLE=<# Zones crossed>-<Fare in dollars>,<# Zones crossed>-<Fare in dollars>… until the maximum number of zones has been reached.

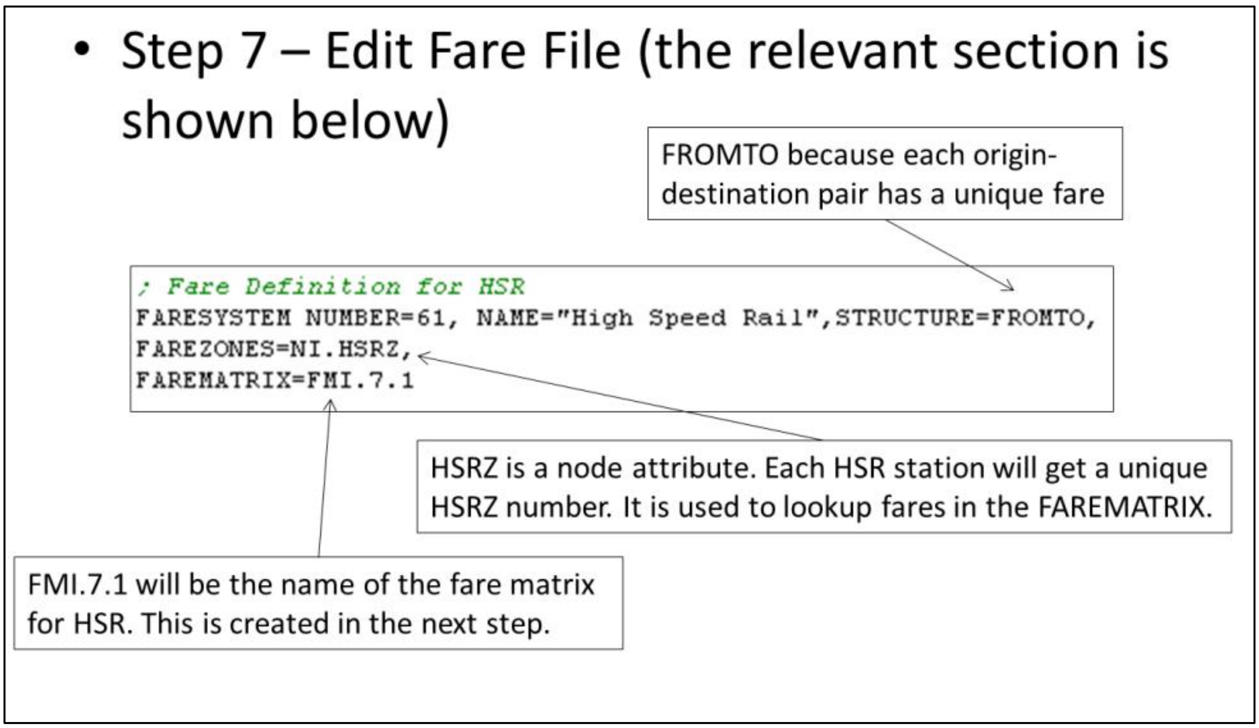


Figure 34: Editing the Fare File for new operators using FROMTO fare structure

If the new operator used a COUNT or FLAT fare structure, the user can stop here. If the new operator uses a FROMTO structure, the next 2 steps need to be executed to create a fare matrix and add this matrix to the CUBE catalog.

Step 8: Create a Matrix File

Figure 35 displays the three steps required to convert the fare matrix created in MS Excel (Step 2) to the format that CUBE requires.

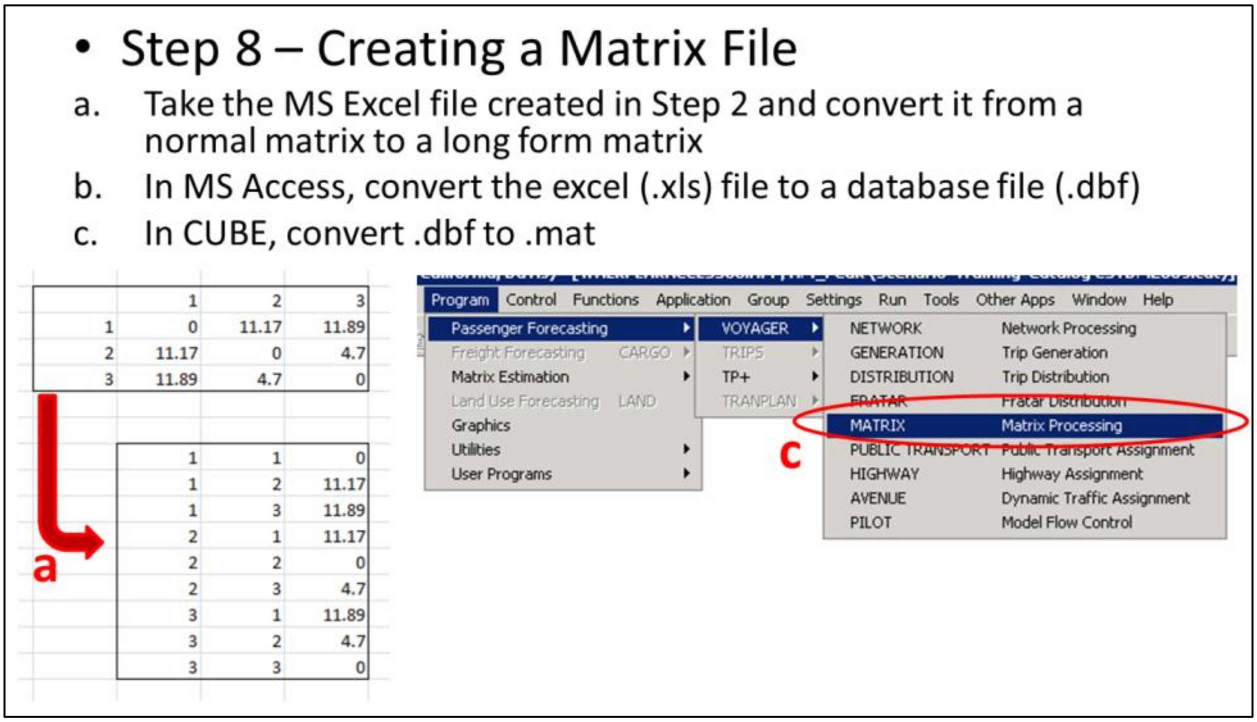


Figure 35: Creating a CUBE Matrix File (.mat)

<u>Step 9: Add a CUBE matrix file to the CUBE catalog</u>

The final step takes the matrix file created in step 8 and adds it to the CUBE catalog. Figure 36 displays the steps needed to accomplish this. Once it has been added, the user should note the Fare Matrix Number that CUBE assigned it (in the example below 1-6 have already been assigned so the next one will be 7) and add this number to the Fare File (Figure 34) in the third line of code.

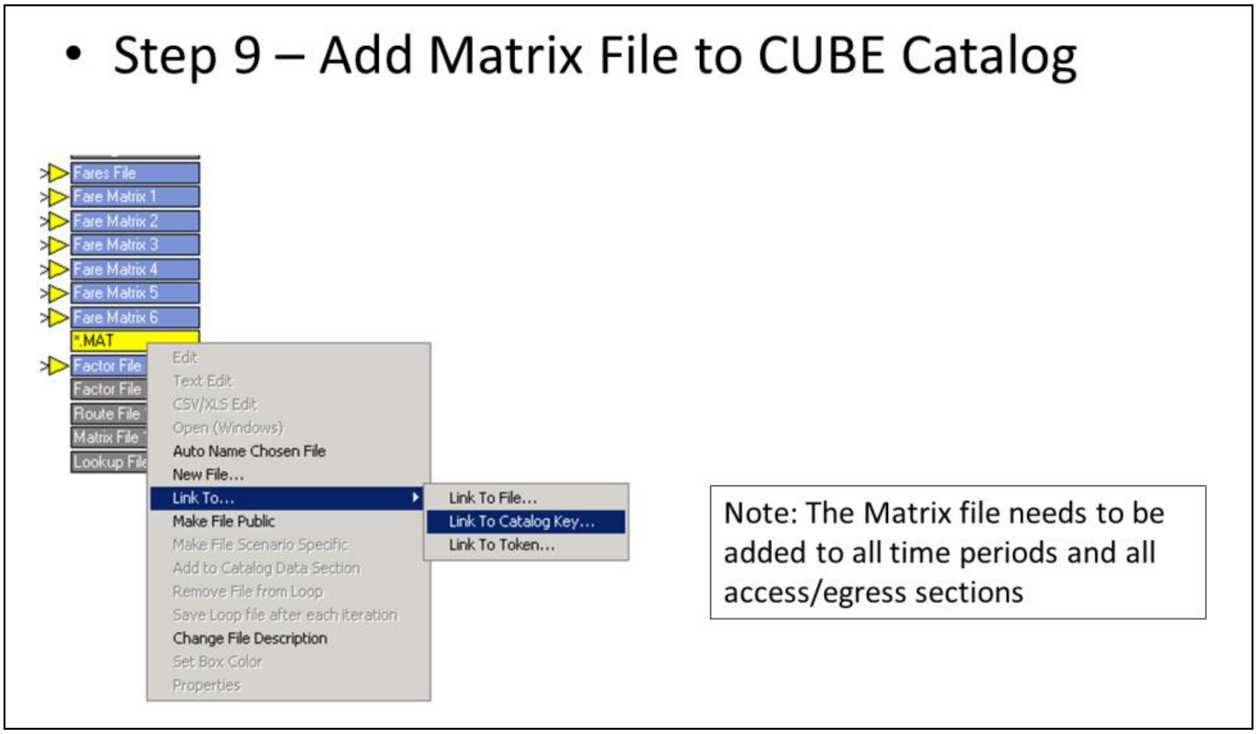


Figure 36: Adding a CUBE Matrix File (.mat) to the CUBE catalog

Appendix C

This part provides an example of the process required to update the local bus system inputs. In the CSTDM, the local bus system is represented through a synthetic methodology that allows for an easier process of update and maintenance of the local transit inputs in the model.

The synthetic methodology for local bus transit computes all information related to the local transit model through four script files (the same for all scenarios). The data inputs for this model are found in the file LocalBus_{year}.dbf, which provides information for catchment areas (transfer and service areas), level of service (LOS) and fares associated to the operators providing service for each TAZ. Figure 37 shows an example of the fields contained in the file LocalBus_{year}.dbf.

If you need to update the information for one or more local bus transit operators, for instance to account for an increase in the quantity of service provided by the operator servicing a specific area, you will need to compute the adjusted LOS for that operator (please refer to the Technical Note on Local Transit Functions for more details). Then, the process of updates will simply require updating the input file LocalBus_{year}.dbf with the updated information for all those TAZs that are included in the service area served by that operator. A similar approach can be followed in case of a change in the fare charged by the local transit operator.

If a new transit operator starts service in an area that previously was not served by local bus transit, you need to introduce a new catchment area in the system. To do this, first determine the TAZs that are served by the new operator (service area) to account for the presence of the new operator (please refer to the Technical Note on Local Transit Functions for the guidelines on how to determine transfer and service areas in CSTDM). Then, you need to update the data input file LocalBus_{year}.dbf, reporting the information for the new catchment area for all TAZs served by the new operator, the information on LOS and fare.

LocalBus_2008

TAZ	COUNTY	OPERATOR	CATCHMENT	LOS	FARE
100	Del Norte	1	1	200	1.12
101	Del Norte	1	1	200	1.12
102	Del Norte	1	1	200	1.12
103	Del Norte	1	1	200	1.12
104	Del Norte	0	0	9999	9999
105	Humboldt	2	2	200	1.35
106	Humboldt	2	2	200	1.35
107	Humboldt	2	2	200	1.35
108	Humboldt	2	2	200	1.35
109	Humboldt	0	0	9999	9999
110	Humboldt	2	2	200	1.35
111	Humboldt	2	2	200	1.35
112	Humboldt	2	2	200	1.35
113	Humboldt	2	2	200	1.35
114	Humboldt	2	2	200	1.35
115	Humboldt	2	2	200	1.35
116	Humboldt	2	2	200	1.35
117	Humboldt	0	0	9999	9999
118	Humboldt	0	0	9999	9999
119	Humboldt	0	0	9999	9999
120	Humboldt	2	2	200	1.35
121	Humboldt	2	2	200	1.35
122	Lassen	0	0	9999	9999
123	Lassen	0	0	9999	9999
124	Lassen	3	3	200	0.75
125	Lassen	3	3	200	0.75
126	Lassen	3	3	200	0.75
127	Lassen	3	3	200	0.75
128	Modoc	0	0	9999	9999

Figure 37: the File LocalBus_{year}.dbf Contains All Relevant Information for the Synthetic Local Bus System

Appendix D

Appendix D provides guidance on setting up the CSTDM to use a number of cores other than the default of 16. Note that each machine setup for a different number of cores is unique to the configuration of that machine, and is based on the number of cores, whether the CPUs are capable of hyperthreading, and the amount of RAM on the system.

The default setup of the CSTDM works well and is tuned for 16 simultaneous threads with 24GB of RAM. Modifying this setup will be subject to some initial guess work followed by testing.

Adjustments to the Demand Models

Setting the demand models (SDPTM, LDPTM, SDCVM) to use an alternate number of cores is a matter of understanding which programming language they use, and the configuration files that manage their settings.

Python Demand Models

The short distance personal travel model(SDPTM) number of threads is controlled by the “num_processes=” setting in the file “settings.py” in the Models\Code folder.. The default CSTDM settings use seven processes, because each of the processes uses approximately 3GB of RAM and increasing the number of processes to the point where all RAM is consumed will cause a system failure. You should also note that the “districts” list (districts = [2,1, 19….]) has been ordered to ensure that the largest and most time consuming districts covering the Bay Area, LA, and San Diego are executed early in the process to avoid having them left for last or have all of the big areas run simultaneously, extending the run time.

It is possible to alter the number of districts used for the SDPTM, but altering the number or contents of the districts requires substantial changes to the scripts that aggregate the results and is not described here.

The number of threads used by the short distance commercial vehicle model is controlled by the nThreads= setting in the text file sdcvm.settings found in \Models\SDCVM\YearXXXX\Inputs. Note that this setting must be changed for each scenario created.

There are fewer restrictions on what this can be set to, but you still want to avoid using all of the system RAM. We found that for the 24GB on our systems 8 threads was the best setting to minimize runtime since we were running it simultaneously with the Java based LDPTM.

Java Demand Model (LDPTM).

The long distance personal travel model is executed in Java and requires that the CUBE script that sets the configurations be edited to change the number of processes. This can be done through CUBE by editing the "`Print PRINTO=1  List="NTHREADS  8"` line in the script(LPMAT00A.S) that creates the control file within the LDPTM program group. To increase or decrease the number of threads just edit the number at the end of the line. In this case 8 threads were used because it was run simultaneously with the SDCVM, which used the remainder of the threads.

Adjustments to the CUBE Cluster Configuration

CUBE Cluster as implemented in the CSTDM requires that you start up a known number of threads and that you assign tasks to them. When making changes to the thread use we recommend carefully diagramming out what each thread will be doing.

It is also important to note that threads can be used as parts of two different multi-processing methods: Intrastep distributed processing (IDP) and Multistep distributed processing (MDP). Given the choice between using a thread for IDP and MDP, MDP will almost always yield greater benefits.

See an example in Figure 38 showing the difference between IDP and MDP. In the figure both the IDP and MDP could use 2 threads. In IDP the first thread completes Program 1, and moves to Program 2, where it encounters the start command for the IDP and hands a portion of the processing to Thread 2. When the combination of

Threads 1 and 2 finishes Program 2, then thread 2 stops and Thread 1 moves onto Program 3.

MDP is more complicated. Thread 1 completes Program 1, then it moves to the Cluster start, and hands off the tasks surrounded by the cluster start and cluster end commands to Thread 2. Then Thread 1 moves on and does Program 3 and then waits at the Pilot step for the signal that Thread 2 is done with Program 2. At the same time Thread 2 is working on Program 2. When that is completed, the cluster group closes, and Thread 2 becomes inactive. When Thread 1 gets the signal that Thread 2 has completed its tasks, it moves on from the Pilot step to Program 4. It's possible that Thread 2 will have completed its tasks very quickly and the notification of its completion will be waiting for Thread 1 when it completes Program 3 and it will not wait at all before starting Program 4.

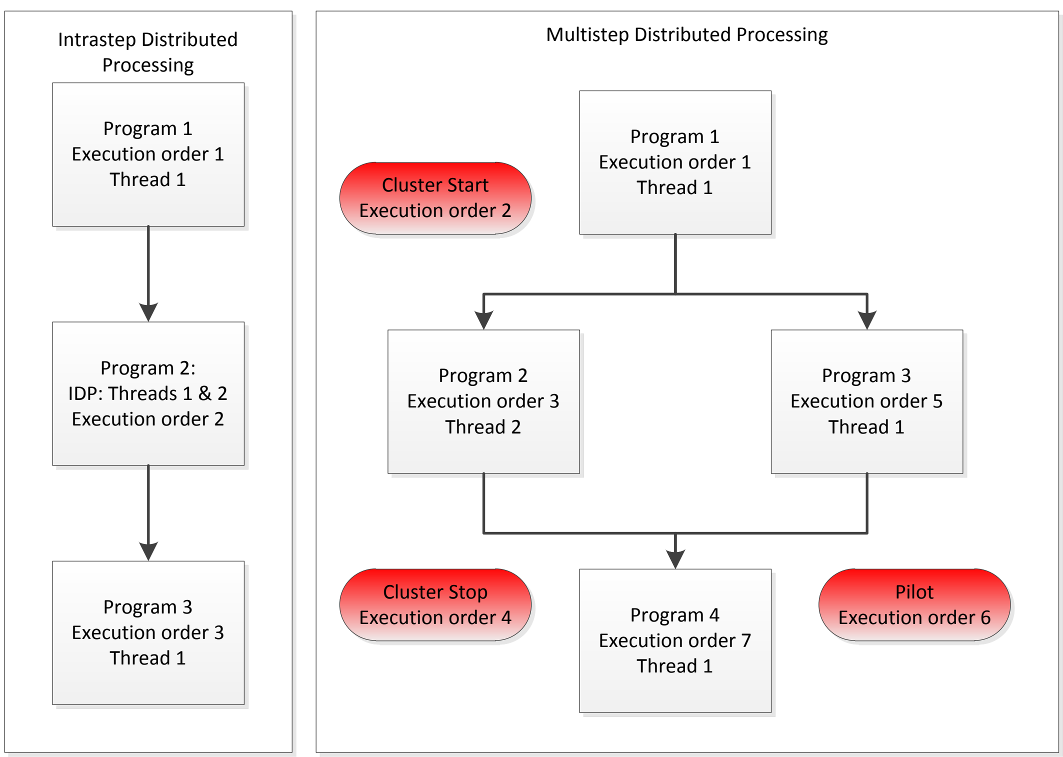


Figure 38: IDP and MDP, both with 2 threads

IDP: *distributes a single set of calculations across multiple processors. This can only be applied to matrix and highway assignment process steps.*

MDP: *distributes sets of independent processing to separate processors letting them run in parallel. Any process step can theoretically be part of an MDP thread.*

Modifications to IDP processes are made within the scripts that call the processes. For example in a highway assignment program, following the FILEI and FILEO statements you will see a statement like:

```
DISTRIBUTEINTRASTEP PROCESSID=cstdm PROCESSLIST=6-9 COMMPATH={CATALOG_DIR}\Cluster\CSTDM\
```

This statement is comprised of the following components:

DISTRIBUTEINTRASTEP: this initiates the intrastep process.

PROCESSID: this is the name of the process id set in the Cluster Node Management settings.

PROCESSLIST: this is the numbers that identify the individual threads that will be used. In the above example threads 6 through 9 would be used. If you also wished to use thread 11 it could be rewritten as “6-9,11”

COMMPATH: is the location where the scripts to be used by each thread are created automatically. This must match the setting in the Cluster Node Management.

This statement is all that is needed to implement IDP for a Highway Assignment or Matrix program in CUBE. Note that the PROCESSLIST contains the threads in addition to the one that originally starts the program.

MDP requires that you insert control programs into the Catalog View. These control the assignment of sets of programs to threads that will be run in parallel, and how the threads will wait for others to complete.

CLUSTER start: Starts a cluster (a set of Programs assigned to a certain thread), an assigns its “manager thread” and the COMMPATH and PROCESSID.

CLUSTER end: Concludes the cluster. Cluster start and end commands are always in matched pairs.

PILOT: includes a script that forces execution of threads to pause when the execution order is reached. This is necessary so that the first thread to finish will wait for other threads to prepare needed input data for a next step. The PILOT will contain a script that looks like:

```
WAIT4FILES  FILES=cstdm1.script.end,cstdm2.script.end, cstdm3.script.end, cstdm4.script.end
```

The Files mentioned in the script are PROCESID + THREADID + ".script.end." The above example will cause the process to halt until threads 1, 2, 3, and 4 have signaled that they have completed their individual tasks.

Needed modifications for Cluster in CSTDM to reduce the number of threads

Many portions of the default setup for the CSTDM use all 16 threads (1 primary plus 15 compute nodes). These are primarily in the Highway Assignment and Public Transport components of the model.

Highway Assignment uses all 16 threads in a combined IDP, MDP setup. IDP is used in both the first and last MATRIX operation. MDP is used to run each time period and IDP is used in the highway assignment within each of the time periods. The number of threads assigned to each highway assignment was balanced iteratively to minimize the total run time by moving threads from the fastest time period to the slowest.

To reduce the number of threads, the easiest would be to remove the IDP threads from highway assignments and rebalance the remaining free threads for IDP.

Public Transport will be complicated to refactor for fewer cores because many portions of the code fit the use of 16 cores very well. There are also several places in the LD Skims section of Public Transport that use IDP, for the most part, those can be eliminated or reduced easily.

The SD Skims section of public transport uses a structure where there are four time periods each of which has four access modes. The combination of these is suited very well to 16 threads since none of the public transport programs can use IDP. To restructure this section to fewer threads our suggestion is that you keep the structure largely intact, but break the time periods so that the AM and PM time periods are run in parallel (MDP) and the Off peak and midday time periods are run following this, also in parallel with each other. Remember that each of the sets processes involved in MDP will take as long as the longest running piece, so it makes sense to run the two time consuming sections together. See Figures 39 and 40. Note that for simplicity in the diagrams, threads are labeled 1-X. In reality you won't see Thread 1, it is the thread that starts all of the processes, and manages the reconnection of all processes. The threads that you are assigning will actually be Threads 2-16, but you'll be calling them as if their number were 1 lower (1-15). The start time is at the top of each figure and the further down each figure the step is, the later in time it is started. All threads wait for the completion of all previously started threads at each pilot step.

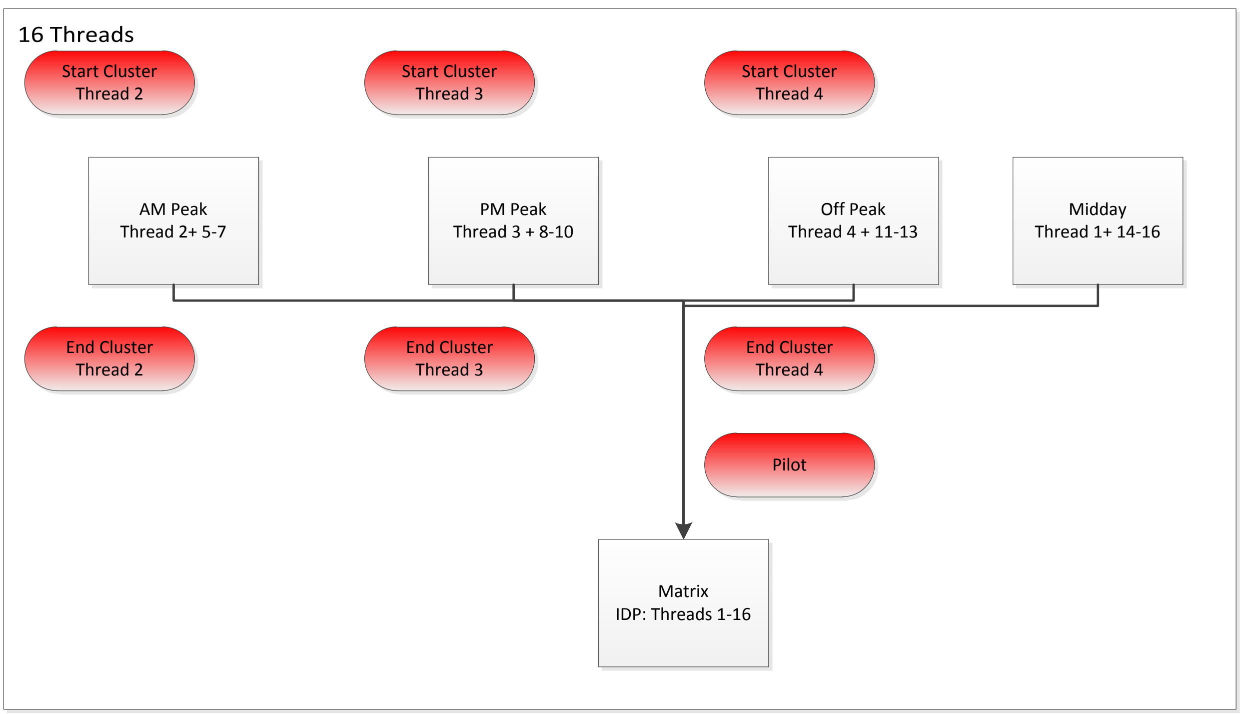


Figure 39: Diagram of flow with 16 threads

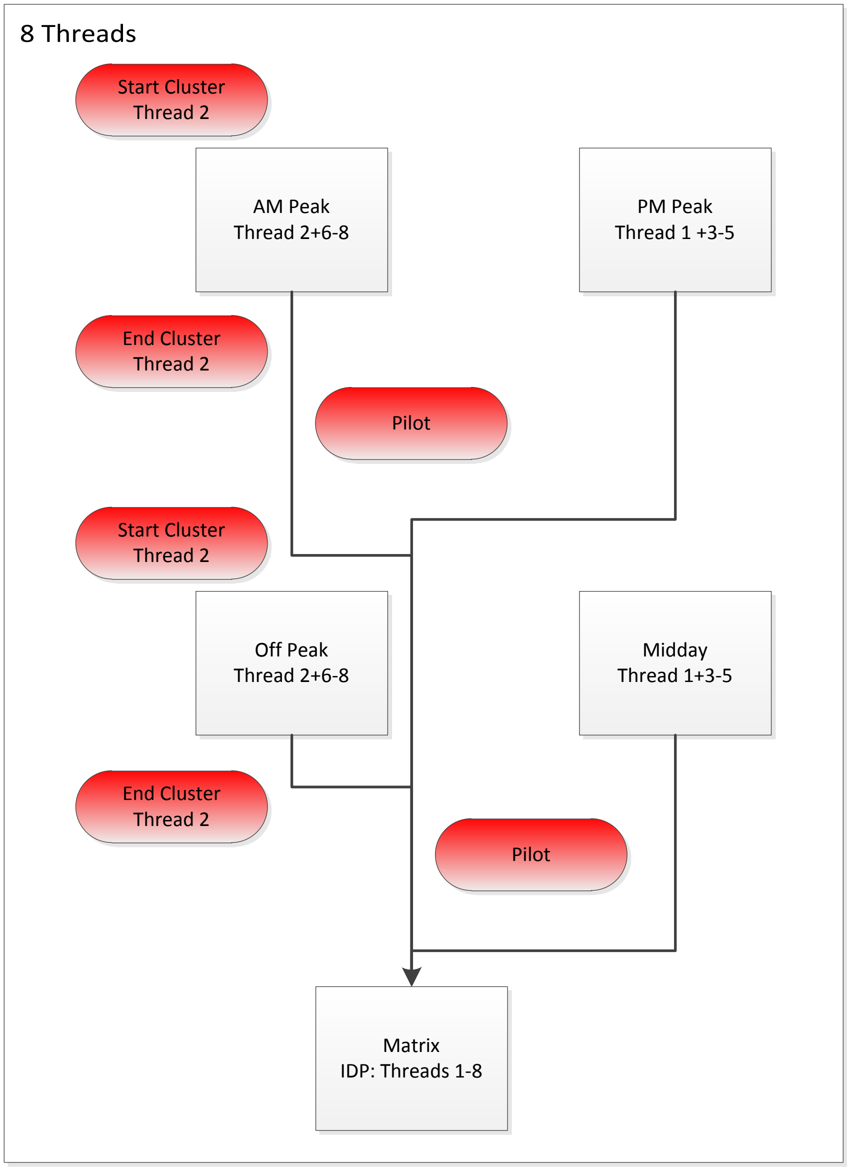


Figure 40: Diagram with 8 threads